



Quelques Techniques de Couplage entre Méthodes Numériques Déterministes et Méthodes de Monte-Carlo

Sylvain Maire

► To cite this version:

Sylvain Maire. Quelques Techniques de Couplage entre Méthodes Numériques Déterministes et Méthodes de Monte-Carlo. Mathématiques [math]. Université du Sud Toulon Var, 2007. tel-00579977

HAL Id: tel-00579977

<https://theses.hal.science/tel-00579977>

Submitted on 25 Mar 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

MEMOIRE

présenté pour obtenir

L'HABILITATION A DIRIGER LES RECHERCHES DE L'UNIVERSITE DU SUD TOULON - VAR

discipline : Mathématiques

Quelques Techniques de Couplage entre Méthodes Numériques Déterministes et Méthodes de Monte-Carlo

par
Sylvain Maire

le 04 décembre 2007

Devant le jury composé de :

M. Benjamin Jourdain	Ingénieur en Chef des Ponts	Rapporteur
M. Christian Lécot	Professeur	Rapporteur
M. Michael Mascagni	Professeur	Rapporteur
M. Khaled Bahlali	Maître de Conférences HDR	Examineur
M. Emmanuel Gobet	Professeur	Examineur
M. Denis Talay	Directeur de Recherche INRIA	Examineur

Chapitre 1

Introduction Générale

Les méthodes de Monte-Carlo [12,46,51,80] constituent une alternative très intéressante aux méthodes déterministes dans la résolution de nombreux problèmes liés aux mathématiques appliquées et à la physique. Elles présentent un grand nombre d'avantages par rapport à ces méthodes qui vont du peu de sensibilité à la dimension et à la géométrie du problème à la facilité de programmation en passant par une parallélisation complètement naturelle. Elles permettent également, via la formule de Feynman-Kac [20,29,30,51], le calcul de la solution approchée de certaines équations aux dérivées partielles linéaires en un point du domaine physique sans avoir à la calculer ailleurs (et donc sans faire de maillage ni résoudre de système linéaire!). L'inconvénient majeur de ce type de méthodes est une vitesse de convergence lente qui ne permet pas d'obtenir un résultat à plus de deux ou trois chiffres significatifs sans un effort calculatoire important. Nous allons rappeler brièvement le principe de la méthode de Monte-Carlo pour le calcul numérique de la valeur de l'intégrale

$$I = \int_D f(x)dx$$

où $D = [0,1]^Q$ avec $Q \in \mathbb{N}^*$ et où $f \in L^2(D)$. Pour en donner une approximation par une méthode de Monte-Carlo, on la réécrit sous la forme $I = E(f(X))$ où X désigne une variable aléatoire Q -dimensionnelle dont les composantes suivent des lois uniformes sur $[0,1]$ et sont indépendantes. On peut alors approcher I par une moyenne empirique

$$\frac{1}{N} \sum_{i=1}^N f(X_i)$$

où les X_i désignent N copies indépendantes de X en utilisant la loi des grands nombres. La vitesse de convergence de cette approximation donnée par le théorème centrale-limite est de

Chapitre 1. Introduction Générale

l'ordre de $\frac{\sigma_f}{\sqrt{N}}$ où σ_f^2 désigne la variance de la variable aléatoire $f(X)$ et vaut donc

$$\sigma_f^2 = \int_D f^2(x)dx - \left(\int_D f(x)dx\right)^2.$$

Une manière d'augmenter la vitesse de convergence est d'essayer de trouver une fonction $\tilde{f}$ qui a même intégrale que f sur D et qui vérifie $\sigma_{\tilde{f}}^2 < \sigma_f^2$. Cette technique dite de réduction de variance [46,49,51,79] peut se mettre en oeuvre en utilisant différentes méthodes dont les plus usuelles sont l'échantillonnage préférentiel, les variables antithétiques, la méthode de stratification ou encore la méthode des variables de contrôle.

Nous allons ici décrire en détail cette dernière afin de bien cibler les enjeux de la réduction de variance. On dispose d'une fonction h dont on sait calculer l'intégrale sur D et telle que h soit une approximation de f , il suffit alors de prendre comme fonction ayant une variance plus faible $\tilde{f}(x) = f(x) - h(x)$. Le calcul de I se fait alors par

$$I = \int_D (f(x) - h(x))dx + \int_D h(x)dx$$

où la première intégrale est calculée par la méthode de Monte-Carlo et où la valeur de la deuxième intégrale est connue exactement. Prenons l'exemple suivant pour illustrer notre propos. On veut calculer l'intégrale

$$I = \int_0^1 \exp(x)dx$$

par la méthode de Monte-Carlo. La variance vaut dans ce cas

$$\sigma_f^2 = \int_0^1 (\exp(x))^2 dx - \left(\int_0^1 \exp(x)dx\right)^2 = 0.2420.$$

Si maintenant on prend $h(x) = x + \frac{x^2}{2}$ comme approximation de f , on doit alors calculer par la méthode de Monte-Carlo l'intégrale

$$J = \int_0^1 \left(\exp(x) - x - \frac{x^2}{2}\right)dx$$

dont la variance vaut cette fois-ci

$$\sigma_{\tilde{f}}^2 = \int_0^1 \left(\exp(x) - x - \frac{x^2}{2}\right)^2 dx - \left(\int_0^1 \left(\exp(x) - x - \frac{x^2}{2}\right)dx\right)^2 = 0.037.$$

Le calcul de I se fait donc par

$$I = J + \int_0^1 (x + \frac{x^2}{2}) dx = J + \frac{2}{3}.$$

La variance a été effectivement réduite puisque l'on ait passé de 0.2420 à 0.037. On aurait pu la réduire encore plus en prenant plus de termes dans l'approximation de la fonction exponentielle.

Cette technique semble donc très intéressante, mais elle présente en fait deux inconvénients qui sont un peu masqués dans cet exemple avantageux. D'une part, le fait de soustraire une fonction h à f exige un coût calcul supplémentaire, celui de l'évaluation de h aux points X_i et cela n'est donc valable que si il est compensé par une réduction suffisante de la variance. D'autre part, l'inconvénient majeur est que la fonction h ait été choisie indépendamment de l'information donnée par les X_i et les $f(X_i)$ et reposait ici sur un développement limité particulier qui n'a aucune chance de s'appliquer dans un cadre général. Ceci est d'ailleurs encore plus vrai pour l'intégration numérique en grande dimension [3,49,57,78] ou pour des problèmes plus complexes liés par exemple aux équations aux dérivées partielles où il n'est pas si facile de trouver des candidats naturels pour la réduction de variance. Pour ses mêmes équations aux dérivées partielles se posent également le problème de la simulation efficace des processus aléatoires qui interviennent dans les représentations probabilistes des solutions de ces équations [25,31,32,41,51,80]. Ceci constitue un travail à part entière qui peut d'ailleurs être couplé avec des techniques de réduction de variance.

D'autres techniques permettent d'utiliser le même principe de moyenne empirique mais en remplaçant les suites purement aléatoires par des suites différentes dites quasi Monte-Carlo [71] ou à discrétion faible qui ont une vitesse de convergence plus rapide que celle donnée par le théorème centrale limite. On peut également citer les techniques de quantification [7,73] qui possède des propriétés analogues bien que basées sur un critère légèrement différent: la distortion. Nous décrirons ceci plus en détail dans la suite.

Les travaux présentés ici s'inscrivent dans ce cadre de la réduction de variance et plus généralement dans l'optimisation de méthodes numériques à l'aide de couplage entre des méthodes déterministes et des méthodes probabilistes. Trois thèmes principaux seront abordés à l'aide de ces techniques: l'intégration numérique sur un hypercube, la résolution d'équations aux dérivées partielles linéaires et le calcul des éléments propres principaux (valeur propre et vecteur propre) de certains opérateurs linéaires.

Chapitre 2

Intégration numérique et approximation

2.1 Introduction

Le calcul numérique d'intégrales [22,49] de fonctions régulières ou non sur un hypercube $D = [0,1]^Q$ a été très largement étudié. Les cas particuliers où des singularités sur les bords du domaine existent ou encore ceux où certaines intégrales doivent être calculées au sens par exemple de la valeur principale de Cauchy requièrent l'utilisation de traitements spécifiques qui sont par exemple décrits dans [49]. Nous en donnerons simplement un exemple comme application de la méthode dite de périodisation décrite ultérieurement. Nous supposons que les fonctions étudiées appartiennent au moins à l'espace $L_2([0,1]^Q)$ mais elles pourront aussi être très régulières. En très grande dimension, la méthode de Monte-Carlo brute apparaît comme la seule possible car sa précision $\frac{\sigma}{\sqrt{N}}$, obtenue en utilisant N tirages, dépend uniquement de la variance $\sigma_f^2 = \int_D f^2(x)dx - (\int_D f(x)dx)^2$ de la fonction à intégrer [22,46,49,78]. Pour le calcul d'intégrales de fonctions très régulières en petites dimensions (de l'ordre de 3 ou 4), les formules de Gauss-produit sont en général considérées comme un bon choix. Toute une gamme de méthodes ont été développées pour des cas intermédiaires entre ces deux extrémités, qu'on peut séparer en deux types principaux.

Le premier type de méthode consiste en l'amélioration de la méthode de Monte-Carlo en utilisant des points mieux répartis dans l'espace que des points purement aléatoires, en rajoutant néanmoins un peu de régularité sur f . Différentes suites dite quasi Monte-Carlo ou encore à discrédance faible [49,71,84] comme les suites de Halton, Faure ou Sobol ont été introduites à cet effet. La qualité de ces suites est donnée par la discrédance qui mesure l'écart entre la distribution des points et la loi uniforme sur D . Ces suites sont très intéressantes en dimension moyenne (de l'ordre de 50) car leur vitesse de convergence pour le

Chapitre 2. Intégration numérique et approximation

calcul d'intégrales est un $O(\frac{\ln(N)^{Q-1}}{N})$. Elles sont de plus relativement faciles à générer. Une autre manière de créer des points mieux répartis dans l'espace est de chercher carrément à optimiser la répartition de ces points suivant certains critères. La quantification [7,73,74] dans le cadre quadratique consiste à chercher les N points minimisant la distortion J définie par

$$J(N) = \min \left(\int_D \inf_{1 \leq i \leq N} |x - x_i|^2 dx : \{x_1, x_2, \dots, x_N \in D\} \right).$$

Une fois ces points calculés à l'aide de l'algorithme de Kohonen, on peut leur associer des poids à l'aide des cellules de Voronoi afin de créer des formules de quadratures. Ces formules s'utilisent pour des fonctions présentant peu de régularité en dimensions modérées.

Une autre catégorie de formules consiste en des formules de type approximation qui sont adaptées spécifiquement à des fonctions présentant une certaine régularité ou une certaine forme. Elles sont inspirées des formules de Gauss tout en essayant de réduire l'effet dimensionnel et peuvent être efficaces pour des dimensions jusqu'à 20. Une première technique est l'utilisation de bases de Fourier Q -dimensionnelle pour les fonctions f périodiques dans toutes les directions et présentant en plus une régularité C^L . En effet, on peut montrer en utilisant des intégrations par parties que les coefficients de Fourier a_m multidimensionnels de ce type de fonctions vérifient

$$|a_m| \leq \frac{C}{(\widetilde{m}_1 \widetilde{m}_2 \dots \widetilde{m}_Q)^L}$$

où $\widetilde{m} = \max(1, |m|)$ et où $L > 1$ est la régularité de f . Ceci permet donc de sélectionner les coefficients significatifs dans l'approximation de f , ceux dont le produit des indices de Fourier est petit. On essaye de construire ensuite des formules d'intégration sur des lattices [43] qui sont exactes pour ce type de coefficients. On définit les croix de Zaremba $Z_{Q,d}$ comme

$$Z_{Q,d} = \{m \in \mathbb{Z}^Q / (\widetilde{m}_1 \dots \widetilde{m}_Q) = d\}.$$

et

$$\rho(V) = \min\{d : V^\perp \cap Z_{Q,d} \neq \emptyset\}$$

pour un certain lattice V . Sloan et Kachoyan [82] ont donné une estimation de la précision de ces formules d'intégration sur des lattices comme un $O(\frac{1}{\rho(V)^L})$. Ces formules sont très efficaces pour des fonctions régulières et périodiques, mais on peut aussi les utiliser pour des fonctions régulières mais pas forcément périodiques. Pour cela, il faut transformer l'intégrale [6] à l'aide d'un changement de variables qui périodise la fonction au sens où la fonction transformée a ses dérivées qui se raccordent jusqu'à un certain ordre aux bornes de l'intervalle d'intégration. Pour cette périodisation, on utilise en pratique le polynôme P de plus bas degré vérifiant

Chapitre 2. Intégration numérique et approximation

$P(0) = 0, P(1) = 1$ et $P^{(i)}(0) = P^{(i)}(1) = 0$ pour $1 \leq i \leq n$. En dimension 1, on écrit alors

$$\int_0^1 f(x)dx = \int_0^1 f(P(t))P'(t)dt$$

de sorte que la fonction $g(t) = f(P(t))P'(t)$ vérifie $g^{(i)}(0) = g^{(i)}(1) = 0$ pour $0 \leq i \leq n-1$ ce qui assure la décroissance voulue. D'autres changements de variables sont également possibles à l'aide par exemple de la fonction tangente hyperbolique. En dimension Q , on effectue la même transformation pour chaque variable d'intégration. Cette technique permet également de gommer certaines singularités (de type $\ln(x)$ ou $\frac{1}{\sqrt{x}}$ en 0) aux bornes de l'intervalle d'intégration. Une étude précise de l'ordre de la méthode dite IMT [42] (qui repose sur l'application de la méthode des trapèzes à la périodisée de f) a été effectuée pour ces deux singularités particulières en dimension un dans [38] et a permis de créer des formules efficaces à la fois pour des fonctions régulières ou faiblement singulières. Ces formules ont ensuite été utilisées avec succès dans la résolution d'équations intégrales de frontières [39] où l'on rencontre simultanément différents types de singularités. L'utilisation d'une quadrature unique pour traiter les différents cas permet une grande simplification des codes de calcul pour une perte de précision faible. L'inconvénient principal de cette méthode de périodisation est que la constante de décroissance C augmente très vite avec n . Ceci est d'autant plus vrai que la dimension Q est grande de sorte qu'en pratique, on choisit en général n égal à 2 ou 3.

Dans le cadre d'approximations polynômiales, Novak et Ritter [72] ont introduit une approche basée sur une combinaison de la formule de Clenshaw-Curtis et de la construction de Smolyak [83]. Elle a permis la construction de formules de quadratures qui sont exactes sur l'espace des polynômes multidimensionnels dont le degré total (c'est à dire la somme des degrés des monômes) est inférieur à une valeur donnée. Leur méthode est très efficace pour des dimensions $Q \geq 8$ et aucune hypothèse de périodicité n'est requise sur f . En utilisant différentes techniques de réduction de variance, on peut aussi construire des méthodes de Monte Carlo possédant des vitesses de convergence accrues. En couplant par exemple la méthode des variables de contrôle à des interpolations polynômiales par morceaux, Atanassov et Dimov [3] ont construit une méthode numérique adaptée aux fonctions dont le degré total de différentiation est fixé. Cette méthode atteint ainsi une vitesse optimale de convergence de $\frac{1}{M^{\frac{1}{2} + \frac{k}{Q}}}$ pour les fonctions réelles ayant toutes leurs dérivées partielles croisées

$$\frac{\partial^r f(x)}{\partial x_1^{\alpha_1} \dots \partial x_Q^{\alpha_Q}}, \alpha_1 + \dots + \alpha_Q = r$$

continues pour $r < k$ et bornées en norme infinie pour $r = k$. Ces méthodes basées sur le degré total permettent de réduire l'effet dimensionnel mais de façon moins drastique que le critère basé sur le produit des indices de Fourier. Elles ont aussi l'inconvénient d'être

optimales uniquement pour un type très précis de fonctions qui n'est pas forcément le plus usuel. Enfin, toutes ces méthodes basées sur des techniques d'approximations ne sont pas bien adaptées aux fonctions très peu régulières. Nous allons présenter dans la suite une série de travaux couplant les techniques d'approximations sur des bases réduites et l'utilisation de points aléatoires, quasi-aléatoires ou issus de la quantification quadratique. Nous aboutirons en bout de compte à des formules de quadratures efficaces simultanément pour des fonctions présentant différents types de régularité en dimension modérée.

2.2 Un algorithme d'approximation

La première étape de notre travail [61,62] a été de créer un algorithme utilisant uniquement des tirages aléatoires permettant le calcul d'approximations quadratiques sur une base orthonormée dont nous rappelons brièvement le principe. Il est fondé sur une utilisation itérative de la méthode des variables de contrôles. Ce type de méthode a été introduite par Halton [37] sous le nom de méthode de Monte-Carlo séquentielle. On cherche à calculer sur $D = [0,1]^Q$ les coefficients $a_k = \langle f, e_k \rangle$ de l'approximation de f sur une base orthonormée de $L^2(D)$ dont les éléments e_k sont bornés. On écrit

$$f(x) = \sum_{k=1}^p a_k e_k(x) + r(x)$$

avec $a_k = \langle f, e_k \rangle$ et $\langle r, e_k \rangle = 0$. Leurs approximations ainsi que celle de f sont obtenus à l'aide d'un échantillon X_i uniforme de taille N par

$$a_k^{(1)} = \frac{1}{N} \sum_{i=1}^N f(X_i) e_k(X_i), \quad f^{(1)}(x) = \sum_{k=1}^p a_k^{(1)} e_k(x).$$

On calcule ensuite une correction

$$b_k^{(1)} = \frac{1}{N} \sum_{i=1}^N (f(Y_i) - f^{(1)}(Y_i)) e_k(Y_i)$$

sur ces coefficients à l'aide de tirages Y_i indépendants des X_i et on obtient une nouvelle approximation des a_k et de f à l'étape 2 par

$$a_k^{(2)} = a_k^{(1)} + b_k^{(1)}, \quad f^{(2)}(x) = \sum_{k=1}^p a_k^{(2)} e_k(x).$$

Chapitre 2. Intégration numérique et approximation

En remplaçant f par son expression et en effectuant M étapes de l'algorithme précédent, on obtient

$$a_k^{(M)} = \sum_{j=1}^p Q_{k,j}^{(M)} a_j + U_k^{(M)}, f^{(M)}(x) = \sum_{k=1}^p a_k^{(M)} e_k(x).$$

Les expressions de $Q_{k,j}^{(2)}$ et $T_k^{(2)}$ sont données par

$$Q_{k,j}^{(2)} = \alpha_{k,j} + \beta_{k,j} - \sum_{s=1}^p \alpha_{s,j} \beta_{k,s}$$

et

$$U_k^{(2)} = R_k^{(1)}(1 - \beta_{k,k}) - \sum_{j \neq k} R_j^{(1)} \beta_{k,j} + R_k^{(2)}$$

avec

$$\alpha_{k,j} = \frac{1}{N} \sum_{i=1}^N e_k(X_i) e_j(X_i), \beta_{k,j} = \frac{1}{N} \sum_{i=1}^N e_k(Y_i) e_j(Y_i)$$

et

$$R_k^{(1)} = \frac{1}{N} \sum_{i=1}^N e_k(X_i) r(X_i), R_k^{(2)} = \frac{1}{N} \sum_{i=1}^N e_k(Y_i) r(Y_i)$$

Le terme $\sum_{j=1}^p Q_{k,j}^{(M)} a_j$ représente l'estimation de a_k si $r(x) \equiv 0$. Le lemme suivant montre qu'il est sans biais et que la variance de ses composantes se réduit géométriquement si on prend suffisamment de tirages par étape.

Lemme 2.2.1. *Il existe des constantes $C(p)$ et $K(p) \leq p^2$, ne dépendant que de la base d'approximation telles que*

$$E(Q_{k,j}^{(M)}) = \delta_{k,j}, \text{Var}(Q_{k,j}^{(M)}) \leq \frac{K(p)^{M-1}}{N^M} C(p)^M.$$

Le terme $U_k^{(2)}$ représente le terme lié à l'erreur de troncature. Le lemme suivant en donne un contrôle en fonction justement de $r(x)$. On aurait le même type de contrôle pour $U_k^{(M)}$.

Lemme 2.2.2. *Il existe des constantes $\gamma(p)$ et $\gamma_1(p)$ telles que*

$$\text{Var}(U_k^{(2)}) \leq 2 \left[\frac{\gamma(p)}{N} + \frac{p^2}{N^2} \gamma_1(p) \right] \int_D r^2(x) dx.$$

Nous avons alors appliqué l'algorithme en dimension un pour des fonctions de classe C^L . Le cadre naturel est une décroissance en $\frac{C}{k^L}$ des a_k qui est vérifiée pour la base de Fourier périodisée et la base des polynômes de Legendre [8]. En effectuant une étude précise des constantes $C(p)$ et $K(p)$, en utilisant cette décroissance et les deux lemmes précédents nous

Chapitre 2. Intégration numérique et approximation

avons montré la convergence de l'algorithme et que la vitesse de convergence des estimateurs obtenus était d'ordre $\frac{1}{N^{L-0.5-\varepsilon}}$ pour l'intégration numérique. Le point crucial pour la suite est la dépendance par rapport à p de la constante $C(p)$. Celle-ci dépend en effet du supremum des intégrales sur le domaine des produits de quatre termes d'éléments de la base d'approximation

$$\left| \int_0^1 e_l(x) e_m(x) e_i(x) e_j(x) dx \right|,$$

où $l, m, i, j \leq p$. Pour la base de Fourier périodisée, cette constante est évidemment bornée par un, indépendamment de p . En revanche, $C(p)$ explose avec p pour la base des polynômes de Legendre. Il est intéressant de trouver une base dans laquelle $C(p)$ n'explose pas et où il n'y a pas les problèmes liés à la périodisation. Un bon compromis semble donc d'utiliser les polynômes de Tchebychef $T_n(x) = \cos(n \arccos(x))$ qui sont les polynômes orthogonaux par rapport au produit scalaire $\langle P, Q \rangle = \int_{-1}^1 \frac{P(x)Q(x)}{\sqrt{1-x^2}} dx$. En effet, grâce à l'équation différentielle

$$\frac{d}{dx}(\sqrt{1-x^2} T_n'(x)) + n^2 \frac{T_n(x)}{\sqrt{1-x^2}} = 0,$$

on peut montrer premièrement, que si $f \in C^{2L}([-1,1])$, les coefficients

$$b_n = \frac{\langle f, T_n \rangle}{\|T_n\|_2^2} = \frac{\int_{-1}^1 \frac{f(x) T_n(x)}{\sqrt{1-x^2}} dx}{\|T_n\|_2^2}$$

de son approximation sur les polynômes de Tchebychef vérifient $|b_n| \leq \frac{C_1}{n^{2L}}$, où C_1 est une constante dépendant de f et L . Ensuite pour calculer les coefficients b_n , on doit modifier légèrement l'algorithme car le calcul Monte-Carlo standard utilisant la loi uniforme conduit à des estimateurs ayant une variance infinie. On écrit donc en utilisant la technique de tirage d'importance $b_n = \frac{E(\pi f(V) T_n(V))}{\|T_n\|_2^2}$ où la densité de V est $\frac{1}{\pi\sqrt{1-v^2}} 1_{[-1,1]}(v)$. Ceci conduit à l'approximation Monte-Carlo

$$b_n \simeq \frac{\pi}{N \|T_n\|} \sum_{i=1}^N f(V_i) T_n(V_i)$$

où les V_i sont N copies indépendantes de V . La simulation de V se fait en écrivant $V = \sin(\pi(U - \frac{1}{2}))$ où U est uniforme sur $[0,1]$. La constante $C(p)$ dépend maintenant des intégrales

$$\left| \int_{-1}^1 \frac{T_l(x) T_m(x) T_i(x) T_j(x)}{\sqrt{1-x^2}} dx \right|,$$

où $l, m, i, j \leq p$, qui sont cette fois-ci bornées par un. La base des polynômes de Tchebychef constitue donc un bon choix en dimension un et nous utiliserons désormais aussi ces

polynômes en dimension supérieure.

2.3 Choix de la base d'approximation multidimensionnelle

Nous supposons dans la suite que $f \in C^{2L}([-1,1]^Q)$. La méthode usuelle d'approximation d'une telle fonction sur la base des polynômes de Tchebychef est d'utiliser le produit tensoriel

$$f(x_1, x_2, \dots, x_Q) \simeq \sum_{i_1=0}^N \sum_{i_2=0}^N \dots \sum_{i_Q=0}^N b_{i_1, i_2, \dots, i_Q} T_{i_1}(x_1) T_{i_2}(x_2) \dots T_{i_Q}(x_Q)$$

où

$$b_{i_1, i_2, \dots, i_Q} = \frac{\langle f, T_{i_1} T_{i_2} \dots T_{i_Q} \rangle}{\prod_{j=1}^Q \|T_{i_j}\|_2^2}.$$

En pratique, on peut remplacer cette approximation par le polynôme d'interpolation $P_N(f)$ aux points de la grille de Tchebychef $y_k = \cos(\frac{2k+1}{N+1} \frac{\pi}{2})$, $k = 0, 1, \dots, N$,

$$P_N(f) = \sum_{i_1=0}^N \sum_{i_2=0}^N \dots \sum_{i_Q=0}^N \alpha_{i_1, i_2, \dots, i_Q} T_{i_1}(x_1) T_{i_2}(x_2) \dots T_{i_Q}(x_Q)$$

où les $\alpha_{i_1, i_2, \dots, i_Q}$ sont définis par

$$\alpha_{i_1, i_2, \dots, i_Q} = \frac{\pi^Q}{\prod_{j=1}^Q \|T_{i_j}\|_2^2 (N+1)^Q} \sum_{j_1=0}^N \dots \sum_{j_Q=0}^N f(y_{j_1}, \dots, y_{j_Q}) T_{i_1}(y_{j_1}) \dots T_{i_Q}(y_{j_Q}).$$

La précision de cette interpolation est décrite précisément dans [8,16]. On a par exemple que

$$\|f - P_N(f)\|_2 \leq \frac{c}{(N+1)^{2L}} \|f\|_{H^{2L}}$$

ce qui montre que cette approximation est de très bonne qualité. Elle est par contre très sensible à l'effet de la dimensionalité car sa complexité est un $O((N+1)^Q)$. Pour atténuer cet effet, nous allons faire une sélection parmi les coefficients de cette approximation en nous inspirant de ce qui a été fait pour la base de Fourier. En posant $\widehat{m} = \max(1, m)$, on peut en effet prouver que

$$|b_m| \leq \frac{C_1}{(\widehat{m_1} \widehat{m_2} \dots \widehat{m_Q})^{2L}}$$

Chapitre 2. Intégration numérique et approximation

en utilisant l'équation différentielle vérifiée par les T_n pour les Q variables d'intégration. On peut alors donner l'approximation

$$f(x_1, x_2, \dots, x_Q) = \sum_{m \in W_{Q,d}} b_m T_{m_1}(x_1) T_{m_2}(x_2) \dots T_{m_Q}(x_Q) + r(x_1, x_2, \dots, x_Q)$$

où l'ensemble

$$W_{Q,d} = \{m \in \mathbb{N}^Q / (\widehat{m_1} \dots \widehat{m_Q}) \leq d\}$$

correspond à un niveau d d'approximation. Le calcul des $L_{Q,d}$ coefficients b_k appartenant à $W_{Q,d}$ se fait en utilisant un programme qui teste si $m_1 m_2 \dots m_Q \leq d$ et stocke les valeurs de ces paramètres dans Q listes $l_1(k), l_2(k) \dots l_Q(k)$. On définit également les constantes de normalisations $c_1(k) = 1_{l_1(k) \geq 1}, \dots, c_Q(k) = 1_{l_Q(k) \geq 1}$ qui interviendront dans la suite. Ces constantes proviennent du fait qu'on a $\|T_0\|_2^2 = \pi$ et $\|T_n\|_2^2 = \frac{\pi}{2}$ si $n \geq 1$. On réécrit alors l'approximation de f comme

$$f(x_1, x_2, \dots, x_Q) \simeq \sum_{k=1}^{L_{Q,d}} b_k T_{l_1(k)}(x_1) T_{l_2(k)}(x_2) \dots T_{l_Q(k)}(x_Q).$$

Afin de donner une idée de la complexité de cette approximation, nous donnons dans le tableau suivant quelques valeurs de $L_{Q,d} = \text{card}(W_{Q,d})$.

d	$L_{2,d}$	$L_{3,d}$	$L_{4,d}$	$L_{5,d}$	$L_{6,d}$
1	4	8	16	32	64
2	8	20	48	112	256
3	12	32	80	192	448
5	21	62	168	432	1072
7	31	98	280	752	1936
10	48	165	504	1432	3872
15	76	276	880	2592	7232

Quelques résultats théoriques concernant la précision de cette approximation et sa complexité, basés sur des arguments simples de théorie des nombres, ont été étudiés dans [63]. On peut donner en particulier un contrôle sur le résidu dans le lemme suivant.

Lemme 2.3.1. *Sous les hypothèses précédentes, $\forall \varepsilon > 0$, il existe une constante positive $C_{Q,\varepsilon}$ dépendant seulement de Q et de ε telle que*

$$\int_D |r(t)|^2 dt \leq \frac{C_{Q,\varepsilon}}{d^{4L-1-\varepsilon}}.$$

Ceci montre que la précision obtenue est du même type que pour les formules d'intégration sur des lattices en perdant un ordre $\frac{1}{2} + \varepsilon$ mais en s'affranchissant des problèmes liés à la périodisation. On perd également un ordre $\frac{1}{2} + \varepsilon$ par rapport à l'approximation classique sur des bases produits mais pour une complexité nettement plus faible.

2.4 Optimisation des points d'intégration

Il reste à trouver des moyens efficaces de calcul des $L_{Q,d}$ coefficients appartenant à l'espace d'approximation. Nous avons commencé par appliquer dans [62,63] notre algorithme basé sur des tirages indépendants de la densité produit

$$w(x) = \prod_{i=1}^Q \frac{1}{\pi \sqrt{1 - x_i^2}} 1_{[-1,1]}(x_i).$$

Cet algorithme est basé sur des corrections successives qui sont liées à des calculs d'intégrales par la méthode de Monte Carlo dont la vitesse de convergence est $\frac{\sigma}{\sqrt{N}}$. On peut donc légitimement envisager de remplacer ces calculs à l'aide d'une méthode analogue mais plus performante. A la place, il est donc très naturel d'utiliser des simulations de cette densité à l'aide de suites à discrétion faible dont la vitesse de convergence est un $O(\frac{\ln(N)^{Q-1}}{N})$. Nous avons expérimenté cette idée dans [64] ce qui a permis d'obtenir une vitesse de convergence environ deux fois plus rapide de l'algorithme. On peut l'expliquer facilement de la manière suivante: si on néglige le terme en logarithme dans la vitesse de convergence, une étape de l'algorithme quasi Monte-Carlo divise l'erreur par un $O(\frac{1}{N})$ alors qu'il en faut deux par la version Monte-Carlo pour obtenir le même résultat. Le fait d'utiliser ces suites quasi Monte-Carlo pour améliorer la vitesse de convergence des méthodes de Monte Carlo dans différents cadres est d'ailleurs assez classique. Ceci a été effectué pour la résolution d'équations différentielles par la méthode de Runge-Kutta [18], pour l'équation de Boltzmann linéarisée [52] ou encore pour la simulation de chaînes de Markov [54].

Cette première amélioration de l'algorithme nous a conduit à proposer une nouvelle version toujours dans [64] de celui-ci en tentant d'utiliser le plus possible l'information donnée par les points où l'on évalue f . Ainsi en utilisant seulement $5L_{Q,d}$ points quasi-aléatoires, nous avons créé un algorithme basé sur des tirages avec remise qui converge dans le calcul des $L_{Q,d}$ coefficients b_n . Cet algorithme ne présentait néanmoins que peu d'intérêt pratique de part sa vitesse de convergence très lente. Ceci nous a amené à nous demander si on peut trouver des manières encore plus efficaces de choisir les points et de les utiliser dans le calcul

des b_n .

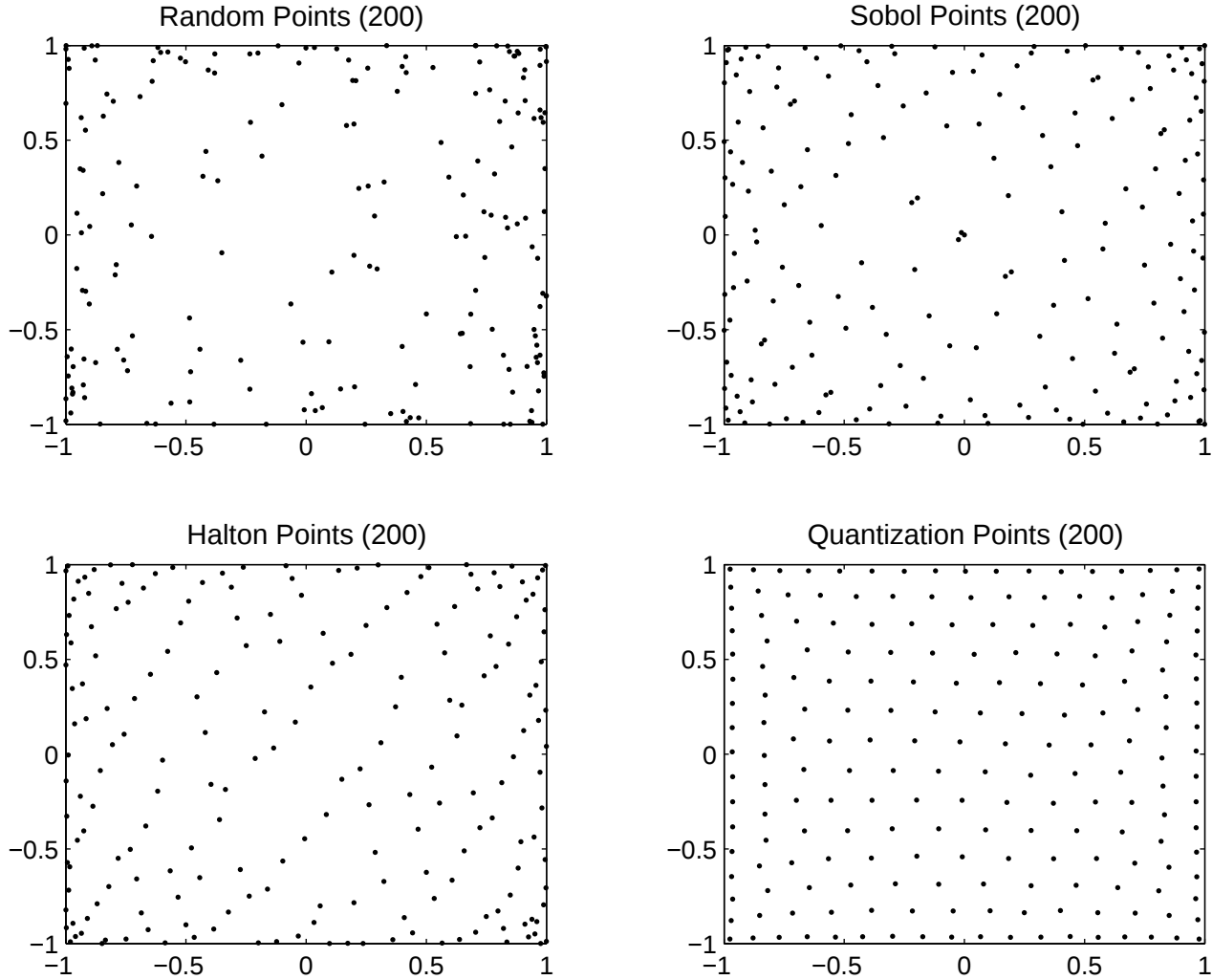
En ce qui concerne le choix des points où l'on évalue f , on peut l'interpréter comme la meilleure façon de représenter la densité de probabilité $w(x)$ à l'aide de M points à partir d'un certain critère. La quantification quadratique optimale [7,73] consiste à trouver les M points dans $D = [-1,1]^Q$ minimisant la fonctionnelle

$$J(M) = \min \left(\int_D \inf_{1 \leq i \leq M} |x - x_i|^2 w(x) dx : \{x_1, x_2, \dots, x_M \in D\} \right).$$

L'utilisation de ce critère quadratique est liée à un algorithme stochastique permettant de résoudre numériquement ce problème de minimisation qu'on peut décrire assez simplement. On commence par tirer M points X_i indépendants qui suivent la densité w . On tire ensuite un autre point Y_1 avec la même densité et on cherche le point X_{min} le plus proche de Y_1 parmi les X_i . On rapproche ensuite X_{min} de Y_1 de telle sorte que sa nouvelle position soit $X_{min} + \varepsilon(Y_1 - X_{min})$ où $\varepsilon > 0$ est un petit paramètre. Le point Y_1 est retiré, on tire un nouveau point Y_2 et on itère le procédé. Le paramètre ε décroît avec le nombre d'étapes n de l'algorithme. Une discussion très complète sur les aspects numériques de l'algorithme dans le cas de variables aléatoires Gaussiennes multidimensionnelles est effectuée dans [74], en particulier sur le choix de ε en fonction de n, M et Q .

On peut également interpréter cet algorithme comme un exemple très abouti de démocratie directe. Le problème de minimisation peut se voir comme un problème de transport optimal où les N points représentent des sources d'énergie que la population, modélisée par la densité w , doit pouvoir atteindre en se déplaçant le moins possible en moyenne quadratique. On place initialement ces sources au hasard suivant la densité w , c'est à dire plus de sources où la population est plus forte. Ensuite, on interroge au hasard (toujours suivant w) une personne de la population et on lui demande ce qu'elle souhaite. La personne demande évidemment à ce que l'on rapproche la source la plus proche de chez elle afin de se déplacer moins pour s'y rendre. On accède à sa requête en rapprochant cette dernière effectivement un peu. On interroge ensuite une autre personne et ainsi de suite tout en diminuant progressivement la distance dont on rapproche la source. Ceci se justifie également en disant que le choix d'une personne ne peut pas remettre en cause complètement le choix de toutes les autres depuis le départ.

Pour donner une idée plus concrète des différentes répartitions de points, nous donnons dans la figure suivante 200 points représentant la densité w en dimension 2 en utilisant respectivement un générateur pseudo-aléatoire linéaire, les suites de Halton, les suites de Sobol et des points de quantification. Il reste maintenant à faire une utilisation globale de l'information donnée par ses points pour calculer les coefficients d'un modèle d'approximation. La technique la plus classique et la plus efficace pour le faire est la méthode des moindres carrés.



Nous allons maintenant décrire la mise en oeuvre de celle-ci dans notre cas et étudier ces différentes propriétés.

2.5 Une méthode de moindres carrés

Notre problème de moindres carrés [9] peut s'écrire comme la minimisation de la fonctionnelle

$$J = \frac{1}{M} \sum_{i=1}^M \left(\sum_{k=1}^{L_{Q,d}} \tilde{b}_k \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}) - f(X_1^{(i)}, \dots, X_Q^{(i)}) \right)^2.$$

Afin d'obtenir une matrice de moindres carrés qui tend vers la matrice identité avec M , on le réécrit comme le problème de moindres carrés pondérés de minimisation de la fonctionnelle

$$J_1 = \frac{1}{M} \sum_{i=1}^M \left(\sum_{k=1}^{L_{Q,d}} \tilde{d}_k \sqrt{2^{\sum_{n=1}^Q c_n(k)}} \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}) - f(X_1^{(i)}, \dots, X_Q^{(i)}) \right)^2$$

avec

$$\tilde{d}_k = \frac{\tilde{b}_k}{\sqrt{2^{\sum_{n=1}^Q c_n(k)}}}.$$

La minimisation de J_1 est équivalente à la résolution du système linéaire $B\tilde{d} = q$ avec

$$B_{k,j} = \frac{\sqrt{2^{\sum_{n=1}^Q c_n(k)+c_n(j)}}}{M} \sum_{i=1}^M \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}) T_{l_n(j)}(X_n^{(i)})$$

et

$$q_k = \frac{\sqrt{2^{\sum_{n=1}^Q c_n(k)}}}{M} \sum_{i=1}^M \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}) f(X_1^{(i)}, \dots, X_Q^{(i)}).$$

La précision d'un problème de moindres carrés dépend du conditionnement de la matrice B et bien sur aussi de l'adéquation au modèle. Ici nous sommes dans une situation favorable car les coefficients $B_{k,j}$ tendent vers δ_{kj} avec M comme ils représentent des calculs d'intégrales par une méthode de Monte-Carlo de produits de fonctions d'une base orthonormée. La situation est d'autant plus favorable que la variance est uniformément bornée par un pour les mêmes raisons que la pour constante $C(p)$. Bien entendu, on peut également calculer ses intégrales en utilisant des suites quasi-aléatoires ou des points de quantification. Nous avons comparé dans [65] ces différentes suites sur des problèmes en dimension trois. Le gain en précision était respectivement d'environ deux chiffres significatifs pour les points de quantification et d'un chiffre significatif pour les suites de Halton par rapport aux points aléatoires. Nous avons néanmoins choisi pour des raisons pratiques liées à la difficulté à construire les points de quantifications de choisir un nombre fixe de $M = 3L_{Q,d}$ points de Halton pour calculer les $L_{Q,d}$ coefficients $\tilde{b}_k$.

2.6 Construction de formules de quadratures

Pour construire des formules de quadratures, on calcule numériquement la matrice inverse B^{-1} et on écrit $\tilde{d} = B^{-1}q$ afin d'obtenir

$$\tilde{d}_k = \sum_{j=1}^{L_{Q,d}} B_{kj}^{-1} q_j = \sum_{j=1}^{L_{Q,d}} B_{kj}^{-1} \frac{\sqrt{2^{\sum_{n=1}^Q c_n(j)}}}{M} \sum_{i=1}^M \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}) f(X_1^{(i)}, \dots, X_Q^{(i)})$$

soit encore

$$\tilde{d}_k = \sum_{i=1}^M \left(\sum_{j=1}^{L_{Q,d}} B_{jk}^{-1} \frac{\sqrt{2^{\sum_{n=1}^Q c_n(j)}}}{M} \right) \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}) f(X_1^{(i)}, \dots, X_Q^{(i)}).$$

On obtient ainsi des formules de quadratures

$$\tilde{b}_k = \sum_{i=1}^M \beta_{i,k} f(X_1^{(i)}, \dots, X_Q^{(i)})$$

pour chacun des $\tilde{b}_k$ en posant

$$\beta_{i,k} = \sqrt{2}^{\sum_{n=1}^Q c_n(k)} \sum_{j=1}^{L_{Q,d}} B_{jk}^{-1} \frac{\sqrt{2}^{\sum_{n=1}^Q c_n(j)}}{M} \prod_{n=1}^Q T_{l_n(k)}(X_n^{(i)}).$$

En intégrant l'approximation de f , on obtient

$$\tilde{I}(f) = \sum_{k=1}^{L_{Q,d}} \tilde{b}_k \prod_{n=1}^Q \int_{-1}^1 T_{l_n(k)}(x) dx = \sum_{i=1}^M \alpha_i f(X_1^{(i)}, \dots, X_Q^{(i)})$$

avec

$$\alpha_i = \sum_{k=1}^{L_{Q,d}} \beta_{i,k} \prod_{n=1}^Q \int_{-1}^1 T_{l_n(k)}(x) dx.$$

Les formules sur $[0,1]^Q$ s'en déduisent immédiatement.

2.7 Résultats numériques

Nous donnons maintenant deux exemples numériques toujours tirés de [65] qui permettent de comparer notre méthode avec les méthodes classiques. Nous commençons par le calcul de

$$I = \int_{[0,1]^6} \exp\left(\frac{x+y+z+t+u+v}{6}\right) dx dy dz dt du dv$$

tiré de [49] page 224. Cette fonction est très régulière et son intégrale vaut $(6(\exp(\frac{1}{6}) - 1))^6 \simeq 1.66$. Le tableau suivant donne l'erreur absolue $|I - \tilde{I}|$ en fonction du nombre $M = 3L_{6,d}$ de points d'intégration.

d	$L_{6,d}$	M	$ I - \tilde{I} $
2	256	768	6×10^{-6}
3	448	1344	8×10^{-7}
5	1072	3216	4×10^{-9}
8	2768	8304	1×10^{-12}

Nous obtenons une précision jusqu' à 12 chiffres significatifs en utilisant 8304 points d'intégration. Si on veut comparer avec les formules d'intégration sur des lattices, on doit

Chapitre 2. Intégration numérique et approximation

périodiser l'intégrale à l'aide de la transformation polynômiale $P(t) = 3t^2 - 2t^3$. La précision obtenue est alors de trois chiffres significatifs en utilisant 10^4 points et quatre avec 4×10^4 points.

Nous considérons maintenant le calcul

$$J = \int_{[0,1]^4} \exp(x) \sin(y) \cos(z) \log(1+t) dx dy dz dt$$

qui a déjà été étudié dans [3,63]. Nous donnons dans le tableau suivant l'erreur absolue $|J - \tilde{J}|$ en fonction du nombre $M = 3L_{4,d}$ de points d'intégration.

d	$L_{4,d}$	M	$ J - \tilde{J} $
1	16	48	2×10^{-2}
3	80	240	5×10^{-5}
6	248	744	7×10^{-6}
17	1041	3123	4×10^{-8}
30	2453	7359	4×10^{-10}

Nous comparons nos résultats avec ceux obtenus par la méthode développée dans [3] qui a un taux de convergence de $\frac{1}{M^{\frac{1}{2} + \frac{k}{Q}}}$ pour les fonctions C_Q^k . Elle est basée sur la combinaison de la méthode des variables de contrôle et d'interpolations polynômiales en $\frac{(Q+k-1)!}{Q!(k-1)!}$ points bien choisis de n^Q cubes. Elle a été utilisée dans le calcul de J avec $Q = k = 4$ et $n = 10$. La précision obtenue est 10^{-8} avec 4×10^5 points d'intégration. Ceci correspond à une approximation numérique de $\frac{-\log(10^{-8})}{\log(4 \times 10^5)} \simeq 1.4$ de l'ordre de la méthode alors que celle de nos quadratures est $\frac{-\log(4 \times 10^{-10})}{\log(7359)} \simeq 2.4$ ce qui est nettement meilleur.

Nous avons étudié d'autres exemples pour des fonctions moins régulières en dimension trois. Ces exemples ont montré que nos formules sont également performantes pour ce type de fonctions au sens où elles sont comparables aux formules de Gauss pour des fonctions deux fois dérivables et plus précises que les suites de Halton pour des fonctions seulement continues.

2.8 Conclusion et perspectives

Après diverses améliorations, nous avons abouti à des formules de quadratures qui ont trois des propriétés requises dans l'article “my dream quadrature rule” [40] de Hickernel. Les mêmes quadratures intègrent très précisément les fonctions très régulières mais aussi de façon satisfaisante les fonctions présentant moins de régularité. Elles sont aussi peu sensibles à l'effet de la dimension. En revanche, elles ne possèdent pas les deux dernières propriétés.

Chapitre 2. Intégration numérique et approximation

Elles sont en effet difficiles à construire et on ne passe pas facilement d'une formule de quadrature à une autre.

La première perspective d'amélioration est de construire des formules de quadrature à grande échelle, c'est à dire pour des dimensions jusqu'à 15 environ, ce qui requiert un gros travail numérique. D'une part, il faut résoudre des systèmes linéaires pleins symétriques de très grande taille. Ensuite, il faut essayer d'optimiser le placement des points ce qui peut passer par exemple par la construction de grilles de quantification avec un très grand nombre de points. La deuxième perspective est l'utilisation de l'approximation sur la base réduite de polynômes de Tchebychef pour résoudre des équations aux dérivées partielles ou encore des équations intégrales en dimension moyenne en s'inspirant de ce qui a été fait par exemple dans [76] pour les équations paraboliques. Un premier travail [27] vient d'ailleurs juste d'être achevé sur la résolution de l'équation de Poisson pour des dimensions jusqu'à 5 en utilisant une formulation variationnelle hybride et aussi l'algorithme de type Monte-Carlo séquentiel qu'on va maintenant décrire dans la partie suivante.

Chapitre 3

Méthodes séquentielles pour les équations aux dérivées partielles linéaires

3.1 Introduction

La formule de Feynman-Kac est un outil très puissant qui permet d'obtenir des représentations probabilistes ponctuelles de nombreuses solutions d'équations aux dérivées partielles comme par exemple les équations de diffusions ou de transport [20,29,30,51]. Si on considère le problème aux limites elliptique avec conditions de Dirichlet dans le domaine $D \subset \mathbb{R}^d$ avec une frontière suffisamment régulière ∂D

$$-Au = f$$

avec les conditions aux limites

$$u = g$$

sur ∂D , la formule de Feynman-Kac s'écrit $\forall x \in D$

$$u(x) = E_x(g(X_{\tau_D}) + \int_0^{\tau_D} f(X_s)ds),$$

où X_t est un processus stochastique solution de l'équation différentielle stochastique relative à l'opérateur A et où τ_D est le temps de sortie de ce même processus du domaine D . Le calcul numérique de cette solution ponctuelle par une méthode de Monte-Carlo conduit à deux types d'erreurs. La première provient de l'erreur de discrétisation de l'équation différentielle stochastique qui intervient dans la simulation numérique du processus X_t . Cette discrétisa-

tion s'effectue en général par le schéma d'Euler de pas de discrétisation Δt qui conduit à une erreur relative sur la solution en $O(\Delta t)$ dans le cas de problèmes dans tout l'espace [4]. Le schéma d'Euler dit naïf en domaine borné ne conduit qu'à une vitesse de convergence en un $O(\sqrt{\Delta t})$. On peut cependant utiliser une approche plus sophistiquée basée sur un pont Brownien [31,32], qui rétablit une erreur en $O(\Delta t)$ en traitant plus en détail ce qui se passe au moment de la sortie du domaine. Si l'opérateur A se réduit au Laplacien sans terme source on dispose de schémas plus efficaces en particulier la marche sur les sphères [25,80]. Une technique récente [41] permet également de traiter des problèmes avec terme source par cette méthode. Dans la partie quatre, nous décrirons plus en détail certains de ces schémas dont nous aurons besoin pour des calculs de lois de temps de sortie. Le second type d'erreur est l'erreur de Monte-Carlo standard en utilisant M simulations c'est à dire $\frac{\sigma}{\sqrt{M}}$ où σ^2 est la variance de

$$g(X_{\tau_D}^x) + \int_0^{\tau_D} f(X_s^x) ds.$$

Il existe différentes techniques pour réduire la variance de ce type de quantités. On peut citer par exemple les travaux de Newton [70] ou encore des algorithmes itératifs de type zéro-variance [10]. Nous allons présenter ici une technique similaire très générale basée sur la méthode des variables de contrôles. Cette méthode va en fait réduire simultanément les deux erreurs jusqu'à une erreur d'approximation sur une base. Ce résultat qui peut paraître surprenant est lié au fait que les deux types d'erreurs ne sont pas complètement découplées. En effet, si f et g ne varient pas beaucoup, même un schéma de discrétisation peu précis peut donner des résultats satisfaisants. Nous allons décrire en détail le fonctionnement de l'algorithme uniquement sur l'équation de Poisson [33]. Diverses généralisations et exemples peuvent être consultés dans [34].

3.2 Description de l'algorithme

La solution au point $x \in D$ de l'équation de Poisson

$$-\frac{1}{2}\Delta u = f$$

dans le domaine $D \subset \mathbb{R}^d$ avec les conditions de Dirichlet

$$u = g$$

sur ∂D est donnée par

$$u(x) = E_x(g(B_{\tau_D}) + \int_0^{\tau_D} f(B_s) ds)$$

Chapitre 3. Méthodes séquentielles pour les équations aux dérivées partielles linéaires

où B_s est un mouvement Brownien d -dimensionnel. A l'aide de cette formule, on peut calculer des valeurs approchées $u_i^{(1)}$ de la solution de cette équation en quelques points x_i en utilisant une méthode de Monte Carlo. En utilisant cette information, on peut ensuite créer une approximation globale et régulière $u^{(1)}(x)$ de la solution de l'équation de Poisson. Ceci peut se faire par exemple par une simple interpolation polynômiale ou en ajustant ces mêmes valeurs à un modèle polynômial. On peut alors utiliser la méthode des variables de contrôle avec $u^{(1)}$ comme approximation de u . On pose $\forall x \in D$

$$y(x) = u(x) - u^{(1)}(x)$$

et on doit maintenant résoudre $\forall x \in D$ l'équation de Poisson

$$-\frac{1}{2}\Delta y = -\frac{1}{2}\Delta(u - u^{(1)}) = f + \frac{1}{2}\Delta u^{(1)}$$

avec les conditions aux limites

$$y = g - u^{(1)}.$$

Une solution globale $y^{(1)}$ peut être alors calculée en utilisant la même méthode que celle utilisée dans le calcul de $u^{(1)}$. Ce procédé est très similaire à celui utilisé dans l'algorithme de la partie 2 pour calculer des intégrales. La variance dans le calcul Monte-Carlo de ces solutions ponctuelles a été réduite. Donc, si on approche maintenant la solution de l'équation initiale par

$$u^{(2)}(x) = u^{(1)}(x) + y^{(1)}(x),$$

on peut légitimement espérer que cette solution soit plus précise que la précédente. Il ne reste plus ensuite qu'à itérer le procédé pour obtenir l'approximation $u^{(n)}(x)$ à la n ème étape de l'algorithme.

3.3 Étude de la convergence

Nous allons maintenant donner les différentes hypothèses et les résultats de convergence de l'algorithme. On suppose que l'approximation de la solution u peut s'écrire sous la forme linéaire

$$Pu(x) = \sum_{j=1}^N u(x_j)\Psi_j(x)$$

pour des fonctions $\Psi_j(x)$. Certaines hypothèses assez faibles sont aussi requises sur le schéma de simulation du mouvement Brownien avec un paramètre de discrétisation Δ . Comme

Chapitre 3. Méthodes séquentielles pour les équations aux dérivées partielles linéaires

l'algorithme est stochastique et biaisé à cause du schéma de simulation, on définit donc

$$m_n = \max_{1 \leq i \leq N} |E(u_n(x_i) - u(x_i))|, v_n = \max_{1 \leq i \leq N} \text{Var}(x_i)$$

pour étudier sa convergence. En ce qui concerne l'influence du schéma de simulation, on considère la différence des solutions des équations de Poisson, avec g à la fois comme terme source et comme condition aux limites, entre les versions discrétisées et continues. Ensuite, on définit $e(g, \Delta, x)$ et $V(g, \Delta, x)$ comme respectivement la valeur moyenne et la variance de la quantité précédente. Nous donnons d'abord le résultat de convergence pour le biais.

Théorème 3.3.1. *Pour tout $n \geq 1$, on a*

$$m_n \leq \rho_m m_{n-1} + \max_{1 \leq i \leq N} |[P(u) - u](x_i) + P[e(u - Pu, \Delta, \cdot)](x_i)|$$

où $\rho_m = \max_{1 \leq i \leq N} [\sum_{j=1}^N |P[e(\Psi_j(\cdot, \Delta, \cdot))](x_i)|]$. Si Δ est suffisamment petit, alors $\rho_m < 1$ et m_n converge à une vitesse géométrique jusqu'à une borne égale à

$$\limsup m_n \leq \frac{1}{1 - \rho_m} \max_{1 \leq i \leq N} |[P(u) - u](x_i) + P[e(u - Pu, \Delta, \cdot)](x_i)|.$$

Ce théorème montre que même si les simulations sont biaisées la borne supérieure sur le biais dépend principalement de la qualité de l'approximation $P(u) - u$. Nous donnons maintenant le résultat de convergence pour la variance v_n .

Théorème 3.3.2. *Posons*

$$C(\Delta, N) = 2 \max_{1 \leq i \leq N} \sum_{j=1}^N \Psi_j^2(x_i) [\sum_{k=1}^N \sqrt{V(\Psi_k, \Delta, x_j)}]^2,$$

$$\rho_v = \max_{1 \leq i \leq N} (\sum_{j=1}^N |P[e(\Psi_j(\cdot, \Delta, \cdot))](x_i)|)^2 + \frac{C(\Delta, N)}{M},$$

alors on a pour tout $n \geq 1$

$$v_n \leq \rho_v v_{n-1} + \frac{1}{M} 2 \max_{1 \leq i \leq N} \sum_{j=1}^N \Psi_j^2(x_i) V(u - Pu, \Delta, x_j) + C(\Delta, N) m_{n-1}.$$

Si Δ est suffisamment petit et M suffisamment grand, alors $\rho_v < 1$ et v_n converge à une vitesse géométrique jusqu'à une borne égale à

$$\limsup m_n \leq \left(\frac{1}{1 - \rho_v} \right) \frac{1}{M} (2 \max_{1 \leq i \leq N} \sum_{j=1}^N \Psi_j^2(x_i) V(u - Pu, \Delta, x_j) + C(\Delta, N) \limsup m_n^2).$$

On peut noter que si $\rho_v < 1$ et $\rho_m < 1$, ce qui est toujours vrai pour Δ suffisamment petit et M suffisamment grand, la convergence a lieu à la fois pour le biais et la variance. Nous allons maintenant donner un exemple d'utilisation de l'algorithme en dimension deux sur un domaine carré.

3.4 Résultats numériques

On considère l'équation de Poisson sur le domaine $D = [-1,1]^2$

$$-\frac{1}{2}\Delta u = -\exp(x+y)$$

avec les conditions de Dirichlet choisies de sorte que la solution de cette équation soit $u(x,y) = \exp(x+y)$. On utilise pour l'approximation des polynômes d'interpolation de Tchebychef bidimensionnel de degré N dans chacune des deux variables x et y . Ceci nécessite donc un calcul de la solution ponctuelle en $(N+1)^2$ points. Dans un souci de réduire les temps de calcul, nous avons utilisé la marche sur les sphères modifiée [41] avec un paramètre d'absorption par la frontière $\varepsilon = 10^{-2}$, ce qui est suffisant pour assurer la convergence. Les erreurs en norme infinie discrète sont mesurées par $e(0)$ et $e(L)$ qui représentent respectivement l'erreur à l'étape 0 (c'est à dire l'erreur Monte-Carlo standard en utilisant M tirages) et l'erreur à convergence.

N	M	L	$e(0)$	$e(L)$	CPU
5	100	8	0.21	7×10^{-5}	1.8
7	300	25	0.23	3×10^{-7}	34
10	1000	40	0.15	8×10^{-11}	540

Les précisions obtenues sont quasiment inaccessibles à une méthode Monte-Carlo standard. Les temps de calculs sont très raisonnables. On remarque cependant qu'ils sont sensibles à l'augmentation du degré des polynômes d'interpolation car il faut prendre plus de tirages pour assurer la convergence. La précision est du même ordre que l'interpolation de la solution exacte et on peut également noter qu'elle est similaire à celle obtenue avec un paramètre $\varepsilon = 10^{-6}$ qui conduisait d'ailleurs à des temps de calculs trois fois plus longs.

3.5 Décomposition de domaines

3.5.1 Introduction

La méthode de Monte Carlo séquentielle précédente nous a permis d'obtenir une approximation spectrale globale de la solution u . Cependant les approximations spectrales sont moins bien adaptées aux solutions moins régulières et la complexité de notre algorithme croît rapidement avec le degré de ces mêmes approximations polynômiales. Les approximations par morceaux sont donc mieux adaptées dans de nombreux cas. Nous allons décrire comment adapter notre algorithme à ce type d'approximations et cela conduira naturellement à une nouvelle méthode de décomposition de domaines. Nous nous limiterons à une décomposition à deux sous-domaines sur l'équation de Poisson. Nous allons commencer par rappeler la façon standard [75] de combiner une méthode déterministe et une méthode de Monte Carlo pour la décomposition de domaines. Le domaine D est divisé en deux sous-domaines D_1 et D_2 séparés par une frontière artificielle $\Gamma_{1,2}$. On commence par calculer une approximation Monte Carlo de la solution de l'équation aux dérivées partielles

$$-\frac{1}{2}\Delta u = f$$

dans D , avec les conditions de Dirichlet

$$u = g$$

sur ∂D , en quelques points x_i de $\Gamma_{1,2}$ en utilisant la formule de Feynman-Kac. En utilisant les valeurs approchées $u_i^{(1)}$ de la solution en ces mêmes points x_i , on peut ensuite construire une approximation globale et régulière $u_{1,2}^{(1)}(x)$ de la solution sur la frontière $\Gamma_{1,2}$ en utilisant par exemple une simple interpolation polynômiale. Cette approximation $u_{1,2}^{(1)}(x)$ est ensuite utilisée comme nouvelle condition aux limites sur $\Gamma_{1,2}$ pour chacun des deux nouveaux problèmes sur D_1 et D_2 . On résout ensuite

$$-\frac{1}{2}\Delta u = f$$

sur chacun des deux domaines D_1 et D_2 en utilisant une méthode déterministe comme par exemple les différences finies ou les éléments finis. Le problème de cette approche est que l'erreur sur l'interface $\Gamma_{1,2}$ est habituellement grande sauf si on utilise un très grand nombre de trajectoires avec un pas de temps petit dans le schéma de discrétisation. Il a été observé dans [75] qu'à l'exception de situations très complexes, le coût calcul d'une approximation précise de l'approximation sur $\Gamma_{1,2}$ est trop grand par rapport à une résolution complète par une méthode déterministe. Des travaux récents [1,2] semblent rendre cette approche plus compétitive en combinant une simulation quasi Monte-Carlo des diffusions [53] et des

techniques de calcul parallèle. Ils ne résolvent cependant pas complètement le problème de précision à l'interface. Nous allons maintenant décrire une technique qui permet de le faire dans le cas de deux sous-domaines et qui a été développée dans [35].

3.5.2 Description de l'algorithme

La première étape est la même qu'avec la méthode standard. En utilisant les valeurs approchées de la solution à quelques points de l'interface, on construit une solution régulière $u_{1,2}^{(1)}(x)$ sur celle-ci. Une fois cette approximation construite, on calcule ensuite des approximations $u_1^{(1)}(x)$ et $u_2^{(1)}(x)$ sur respectivement D_1 et D_2 en utilisant toujours une méthode de Monte-Carlo et des techniques d'interpolation comme dans l'algorithme de la section 3.2.

Comme la solution globale u doit être régulière, les approximations sur D_1 et D_2 doivent vérifier certaines conditions d'interface. La méthode déterministe dite du complément de Schur [28,58] donne une solution globale qui est continue et a des dérivées normales continues à la frontière $\Gamma_{1,2}$. Les approximations $u_1^{(1)}(x)$ et $u_2^{(1)}(x)$ ne vérifient pas ces conditions cruciales. Pour assurer ces conditions, on transforme ces approximations sur chaque sous-domaine grâce à l'erreur $h(x) = u_1^{(1)}(x) - u_1^{(2)}(x)$ sur $\Gamma_{1,2}$. Par exemple, si $D =] - 1, 1[, D_1 =] - 1, 0[,$ et $D_2 =] 0, 1[$ les nouvelles approximations deviennent respectivement $u_1^{(1)}(x) - \frac{h'(0)g(x)+h(0)}{2}$ sur D_1 et $u_1^{(2)}(x) + \frac{h'(0)g(x)+h(0)}{2}$ sur D_2 . La fonction g doit vérifier $g(0) = 0$ et $g'(0) = 1$. Un choix naturel est la fonction $g_1(x) = x$, mais son utilisation perturbe les nouvelles solutions $u_1^{(1)}$ et $u_2^{(2)}$ sur chaque sous-domaine loin de l'origine. Pour atténuer ce problème, on peut utiliser par exemple les fonctions

$$g_{2,n} = x(1+x)^n 1_{[-1,0]}(x) + x(1-x)^n 1_{[0,1]}(x)$$

qui sont très proches de zéro quand on s'éloigne de $x = 0$.

On résout ensuite la nouvelle équation de Poisson

$$-\frac{1}{2}\Delta y_{1,2}^{(1)} = -\frac{1}{2}\Delta(u - u^{(1)}) = f + \frac{1}{2}\Delta u^{(1)}$$

avec les conditions aux bords

$$y_{1,2}^{(1)} = g - u^{(1)}$$

à quelques points de l'interface $\Gamma_{1,2}$ et on pose $y_{1,2}^{(1)}(x)$ l'approximation de ce résidu. La dernière étape est de résoudre sur chaque sous-domaine les équations

$$-\frac{1}{2}\Delta v^{(1)} = -\frac{1}{2}\Delta(u - u_i^{(1)}) = f + \frac{1}{2}\Delta u_i^{(1)}$$

Chapitre 3. Méthodes séquentielles pour les équations aux dérivées partielles linéaires

avec les conditions aux limites

$$v^{(1)} = y_{1,2}^{(1)}$$

sur l'interface $\Gamma_{1,2}$ et

$$v^{(1)} = g - u_i^{(1)}$$

sur les autres bords. La nouvelle approximation $u^{(2)}(x)$ est la régularisation de la fonction

$$u^{(1)}(x) + v^{(1)}(x)$$

à l'aide du procédé précédent. On itère ensuite l'algorithme jusqu'à convergence.

3.5.3 Résultats numériques

Nous reprenons le même exemple qu'à la section 3.4 sur le domaine $D =]-1,1[\times]0,1[$ divisé en deux domaines carrés $D_1 =]-1,0[\times]0,1[$ et $D_2 =]0,1[\times]0,1[$. Les solutions ponctuelles sont calculées à l'aide de la marche sur les sphères avec un paramètre $\varepsilon = 5 \times 10^{-3}$. Les approximations sur chaque domaine utilisent des interpolations polynômiales bidimensionnelles de type Tchebychef de degré N et une interpolation unidimensionnelle de même degré est utilisée sur la frontière. Nous utilisons la fonction $g_{2,4}$ dans le procédé de régularisation quand $N = 5,7$ et la fonction $g_{2,6}$ quand $N = 9$ qui est réellement plus efficace dans ce cas précis.

N	M	L	$e(0)$	$e(L)$	CPU
5	100	21	0.18	3×10^{-6}	1.9
7	400	39	0.05	2×10^{-9}	32
9	1000	52	0.07	2×10^{-12}	148

Comme pour la méthode purement spectrale, on atteint une très grande précision sur les solutions. La gestion de l'interface a donc été bien effectuée. La majorité des trajectoires que l'on simule sont localisées dans les sous-domaines. Seuls les calculs des solutions à l'interface utilisent des trajectoires qui vivent dans tout le domaine. De plus le fait d'utiliser des approximations locales permet de diminuer le degré des polynômes d'interpolation et donc de réduire les temps-calculs.

3.6 Conclusion et perspectives

La méthode présentée ici a été validée sur différents types de problèmes et des preuves de convergence ont été effectuées dans le cas d'approximations globales pour toute une classe

Chapitre 3. Méthodes séquentielles pour les équations aux dérivées partielles linéaires

d'équations aux dérivées partielles linéaires. Une extension de la méthode à des approximations locales a conduit à une nouvelle version Monte-Carlo des méthodes de décomposition de domaines. Il reste à effectuer les preuves de convergence de cette méthode et aussi son extension à plusieurs sous-domaines.

De manière plus générale, de nombreuses voies restent à creuser aussi bien dans l'optimisation de cette méthode que de son extension à d'autres problèmes plus compliqués afin de la rendre vraiment concurrentielle avec les méthodes déterministes. Du point de vue de l'optimisation, on peut penser à des techniques de calcul parallèle couplé ou non à la décomposition de domaines ainsi qu'à des schémas de simulation des processus plus sophistiqués. En particulier, comme certains points comme les points de Tchebychef sont fixés dans les simulations, on pourrait sans doute quantifier une fois pour toutes les trajectoires issus de ces points particuliers. Un travail en collaboration avec Etienne Tanré [67] est en cours sur ce sujet précis et sur la création de formulations spectrales stochastiques. D'un point de vue extension, on peut penser naturellement à des problèmes en dimension plus grande ou encore à des opérateurs sous forme divergence. Cela conduira à l'étude de nouveaux schémas de discrétisations des processus ainsi qu'à celle de nouvelles techniques d'approximations des fonctions comme par exemple celles de la partie 2.

Chapitre 4

Approximation de la valeur propre principale de différents opérateurs linéaires

4.1 Description de la méthode

Le calcul des valeurs propres par une méthode déterministe d'opérateurs linéaires, comme le Laplacien ou des opérateurs de transport, exige tout d'abord une discrétisation de celui-ci à l'aide par exemple de schémas comme les différences finies ou les éléments finis. Il faut ensuite calculer les valeurs propres de la matrice de discrétisation via une méthode numérique appropriée. Enfin, il faut s'assurer que cette discrétisation a été suffisamment fine pour que les valeurs propres de la matrice soient suffisamment proches de celle de l'opérateur. Ceci peut être très coûteux quand la solution vit dans un espace de grande dimension. Dans de nombreux cas, seul un petit nombre de valeurs propres sont intéressantes d'un point de vue physique. En particulier, la valeur propre dite principale de l'opérateur A (celle de plus grande partie réelle) joue un rôle crucial dans de nombreuses situations. Elle détermine en effet l'évolution temporelle de la solution $u(x,t)$ du problème de Cauchy

$$\frac{\partial u}{\partial t}(x,t) = Au(x,t)$$

avec une donnée initiale u_0 . On peut écrire dans le cas où la valeur propre principale α est simple que

$$u(x,t) = C_0 e^{\alpha t} + C_1 e^{\alpha_1 t} + \dots + C_k e^{\alpha_k t} + o(e^{\alpha_k t})$$

où C_0 ne dépend pas de t , où les autres C_i croissent moins vite qu'un polynôme au voisinage de l'infini en temps et où ce développement est valable presque partout. Dans le cadre de

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

la neutronique, on s'intéresse au signe de α qui détermine si le réacteur est sous-critique ou surcritique. Si A est un opérateur de type diffusion comme le Laplacien avec des conditions de Dirichlet et éventuellement un terme source supplémentaire, la valeur propre principale détermine la vitesse de convergence temporelle vers la solution stationnaire. On peut écrire de plus si C_0 est strictement positif, c'est à dire si la composante au point x de la projection de u_0 sur le premier vecteur propre est non nulle, que

$$\frac{1}{t} \log(u(x,t)) \approx \frac{1}{t} \log(C_0 e^{\alpha t} + C_1 e^{\alpha_1 t})$$

en ne gardant que les deux termes prépondérants. On a donc

$$\frac{1}{t} \log(u(x,t)) \approx \frac{1}{t} \log(C_0 e^{\alpha t} (1 + \frac{C_1}{C_0} e^{(\alpha_1 - \alpha)t}))$$

soit à l'aide d'un développement limité au premier ordre que

$$\frac{1}{t} \log(u(x,t)) \approx \alpha + \frac{\log(C_0)}{t} + \frac{C_1}{C_0} \frac{e^{(\alpha_1 - \alpha)t}}{t}.$$

L'expression précédente prouve donc que presque partout

$$\lim_{t \rightarrow \infty} \frac{1}{t} \log(u(x,t)) = \alpha$$

mais surtout précise de quelle façon a lieu cette convergence temporelle et va permettre de proposer différentes façons d'approcher α à l'aide des valeurs de u à différents instants. Pour cela, il faut être capable de calculer ces quantités à l'aide d'une méthode numérique adaptée. C'est justement le cas de la méthode de Monte-Carlo appliquée aux processus de transport ou de diffusion qui permet le calcul de u à différents instants à l'aide des mêmes simulations pour des problèmes de grande dimension via la formule de Feynman-Kac qui s'écrit

$$u(x,t) = E(1_{\tau_D^x > t} u_0(X_t) \exp\left(\int_0^t c(X_s) ds\right)).$$

Le processus X_t est sous-jacent à l'équation aux dérivées partielles considérée, τ_D^x est le temps de sortie du domaine issu de x et c est une fonction, incluse ou non dans l'opérateur A , qui représente un terme de gain ou de perte suivant son signe. L'approximation de $u(x,t)$ se fait par la méthode de Monte-Carlo standard. En utilisant N trajectoires indépendantes qui partent de x , on a

$$u(x,t) \simeq \frac{1}{N} \sum_{i=1}^N Z_t^{(i)}$$

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

où les $Z_t^{(i)}$ sont N copies indépendants de la variable aléatoire

$$Z_t = 1_{\tau_D^x > t} u_0(X_t) \exp \left(\int_0^t c(X_s) ds \right).$$

On voit donc qu'on peut calculer à l'aide de cette formule la solution $u(x, t)$ à différents instants t_i en utilisant les mêmes trajectoires. Nous aurons de plus à calculer la valeur moyenne

$$g_E(t) = \frac{1}{\text{vol}(E)} \int_E u(x, t) dx$$

sur une partie E du domaine D . La seule différence dans le calcul de son approximation Monte-Carlo $\bar{g}_E(t)$ est que le point de départ des trajectoires est tiré uniformément dans E . Nous avons développé plusieurs approches pour calculer α à partir de son développement spectral. L'approche la plus efficace et qui utilise de manière globale l'information donnée par le calcul de la solution à différents instants consiste à utiliser une méthode de moindres carrés pour calculer α et $\log(K_0)$ en ajustant ces paramètres au modèle d'approximation en négligeant le terme $\frac{K_1}{K_0} \frac{e^{(\alpha_1 - \alpha)t}}{t}$. On doit donc trouver α et $\beta = \log(K_0)$ minimisant

$$\sum_{i=p}^q \left(\alpha + \frac{\beta}{t_i} - \frac{1}{t_i} \log(g_E(t_i)) \right)^2,$$

ce qui constitue un problème de moindres carrés linéaire par rapport à α et β , et qui conduit à des estimateurs $\alpha(t_p, t_q, m)$ de α sur l'intervalle $[t_p, t_q]$ qu'on a découpé en m morceaux. La principale difficulté consiste à se rendre compte numériquement à partir de quel instant le fait de négliger le terme $\frac{K_1}{K_0} \frac{e^{(\alpha_1 - \alpha)t}}{t}$ est légitime. Pour ce faire, différentes techniques liées aux choix des intervalles de temps, aux nombres d'instants m et à des critères d'adéquation des estimateurs au modèle ont été développées dans [60,66] puis affinées dans [55,56] pour obtenir des approximations les plus précises et les plus robustes possibles.

4.2 Calcul critique en neutronique

4.2.1 Description du problème

L'évolution de la population de neutrons est modélisée par une fonction de densité $u(x, v, t)$ qui dépend de la position x , de la vitesse v et du temps t dans un domaine $D = X \times V \subset \mathbb{R}^3 \times \mathbb{R}^3$. Cette fonction de densité est solution du problème de Cauchy

$$\frac{\partial u}{\partial t}(x, v, t) = Tu(x, v, t)$$

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

où l'opérateur de transport neutronique T est défini par

$$Tu(x,v,t) = -v\nabla_x u(x,v,t) - \Sigma_t(x,v)u(x,v,t) + \int_V \Sigma(x,z,v)u(x,z,t)dz + S(x,v,t).$$

$S(x,v,t)$ représente un terme source, $\Sigma_t(x,v)$ et $\Sigma(x,v)$ sont la section efficace totale et un terme incluant la section efficace de diffusion Σ_s et la section efficace de fission Σ_f . Ces quantités caractérisent les propriétés physiques du réacteur nucléaire qui est habituellement très hétérogène [15]. En se donnant une condition initiale positive $u_0(x,v) = u(x,v,0)$ et des conditions aux limites, on peut alors calculer la solution positive $u(x,v,t)$. Ces conditions aux limites peuvent être de type absorption ou réflexion. On s'intéresse ici au problème plus spécifique

$$\frac{\partial u}{\partial t}(x,v,t) = T_\lambda u(x,v,t)$$

dit problème de criticité pour l'équation de transport

$$T_\lambda u(x,v,t) = -v\nabla_x u(x,v,t) - \Sigma(x,v)u(x,v,t) + \int_V f_s(x,z,v)u(x,z,t)dz + \frac{1}{\lambda} \int_V f_f(x,z,v)u(x,z,t)dz$$

avec $\lambda > 0$ et où la production de neutrons vient seulement de la fission. Le rôle de λ appelé aussi *keff* est de contrôler le nombre de neutrons créés par une fission. On doit calculer la valeur de λ telle que la valeur propre principale α_λ de l'opérateur T_λ soit la plus proche de zéro que possible. Si $\alpha_\lambda = 0$, il y a alors un parfait équilibre entre la production et l'absorption de neutrons. Si $\alpha_\lambda < 0$, le réacteur est dit sous-critique et si $\alpha_\lambda > 0$, le réacteur est surcritique. Le nombre de neutrons décroît ou augmente exponentiellement vite suivant le signe de α_λ . De plus, sous des hypothèses relativement faibles, α_λ est une fonction croissante de λ qui tend vers $+\infty$ quand $\lambda \rightarrow 0$ et $\alpha_\lambda < 0$ quand λ est suffisamment grand. L'existence et l'unicité de la valeur propre principale et d'une fonction propre associée positive pour le calcul de α sont étudiés dans Mika [68]. La même étude pour le calcul de λ est faite dans Vidav [85]. On peut également montrer que sous des hypothèses faibles sur le modèle physique, en utilisant des outils d'analyse [69] et principalement le théorème de Krein-Rutman que cette valeur propre principale est réelle et simple. Finalement, on peut trouver un résumé de ces résultats et des développements sur l'approximation d'équations de transport par des équations de diffusion dans Bardos [5]. Les méthodes numériques déterministes les plus utilisées pour le calcul du *keff* sont les méthodes nodales, les différences finies et les éléments finis pour la discrétisation spatiale les méthodes SN et PN pour les variables de vitesses Lewis [59], Planchard [77] Warin [86]. Bien entendu, ces méthodes sont très coûteuses surtout pour les problèmes en dimension 3. Pour ces problèmes difficiles, on peut aussi considérer des

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

méthodes de Monte-Carlo de type particulière Lapeyre-Pardoux-Sentis [51] et Kalos [45] aussi bien que pour le calcul de la valeur propre principale que pour le calcul du *keff* [13]. Ces méthodes reposent cependant comme les méthodes déterministes sur un calcul de la solution sur tout le domaine, contrairement à ce qui est proposé ici.

4.2.2 L'interprétation probabiliste

Les neutrons se déplacent dans un domaine borné où ils collisionnent à des instants aléatoires avec des particules du milieu. Le mouvement d'un neutron est décrit à l'aide d'un processus stochastique X_t solution de l'équation différentielle $\frac{dX_t}{dt} = b(X_t, Y_t)$ où Y_t est un processus de saut qui modélise la vitesse de la particule [20]. L'évolution de cette vitesse est décrite par une loi de probabilité $\Pi^{x,y}(dz)$ qui dépend uniquement de x et y telle que $\forall z$, $\Pi^{x,y}(\{z\}) = 0$. La loi de la vitesse Y_t après une collision sachant que $(X_t, Y_t) = (x, y)$ juste avant celle-ci est donnée par $\Pi^{x,y}(dz)$. Nous décrivons maintenant la fréquence des collisions avec le milieu. On définit T_n comme le temps entre deux sauts successifs aux temps t_n et t_{n+1} : $T_n = t_{n+1} - t_n$. La loi conditionnelle de T_n sachant que $(X_t, Y_t) = (x, y)$ au temps t_n est donnée par sa fonction de répartition qui est indépendante de n :

$$F^{x,y} = 1 - \exp \left[- \int_0^t \theta(X_s(x, y), y) ds \right]$$

où $X_s(x, y)$ est solution de $\frac{dX_t}{dt} = b(X_t, Y_t)$ avec $X_0 = x$ et $Y_0 = y$. La fonction θ est positive et bornée. La représentation probabiliste de la solution du problème de Cauchy dans un domaine D

$$\begin{aligned} \frac{\partial u(x, y, t)}{\partial t} &= b(x, y) \nabla_x u(x, y, t) + c(x, y) u(x, y, t) + \\ &\theta(x, y) \left[\int_V u(x, z, t) \Pi^{x,y}(dz) - u(x, y, t) \right] \end{aligned}$$

avec des conditions absorbantes et comme condition initiale

$$u(x, y, 0) = u_0(x, y)$$

est donnée par la formule de Feynman-Kac par

$$u(x, y, t) = E_{x,y} \left[1_{\tau_D^{x,y} > t} u_0(X_t, Y_t) \exp \left(\int_0^t c(X_s, Y_s) ds \right) \right]$$

où $\tau_D^{x,y}$ est le temps de sortie de D en partant de (x, y) et V l'espace des vitesses. Cette formule est vraie si on suppose par exemple que b est mesurable et Lipschitzienne en x ,

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

uniformément en y et que c et u_0 sont des fonctions mesurables bornées. Y_t est un processus de saut sur V . X_t est solution de l'équation différentielle $\frac{dX_t}{dt} = b(X_t, Y_t)$.

4.2.3 Résultats numériques

Nous avons étudié dans un premier temps plusieurs problèmes homogènes spatialement tirés de [19]. Pour ces problèmes, la formule de Feynman-Kac se simplifie énormément si on suppose que la condition initiale $u_0(x, y)$ vaut identiquement 1. On a alors simplement

$$u(x, y, t) = \exp(ct) P(\tau_D^{x, y} > t)$$

de sorte que la seule quantité à calculer par une méthode de Monte-Carlo est la fonction de répartition du temps de sortie du processus de transport à différents instants. Ce choix particulier de u_0 se justifie également du fait qu'il constitue une bonne approximation de la fonction propre associée à la valeur propre nulle. Ceci permet de rendre le terme $C_0 e^{at}$ plus rapidement dominant et ainsi d'éviter d'avoir à calculer pour des temps trop grands $P(\tau_D^{x, y} > t)$, ce qui conduit à des calculs de probabilités d'événements rares. Les résultats numériques ont validé notre approche en particulier la méthode de moindres carrés au sens où nous avons obtenu une précision de 5 chiffres significatifs aussi bien sur la valeur propre principale que sur le facteur critique pour tous les modèles testés. Une autre technique de réduction de variance basée sur des processus de branchement a été introduite dans [56].

Nous avons étudié ensuite des modèles numériques plus complexes en dimension deux ou trois issus de problèmes donnés par l'EDF [86]. Ces modèles sont inhomogènes, pas forcément isotropes et le calcul de la valeur propre ne peut donc plus se faire en calculant uniquement la fonction de répartition du temps de sortie du domaine D , mais en calculant vraiment $u(x, y, t)$. De fait de cette inhomogénéité, la simulation du processus Markovien de saut exige un suivi précis de la trajectoire tenant compte de la géométrie et de la nature différente suivant les zones des lois exponentielles nécessaires à la simulation des chocs. Ceci s'effectue à l'aide d'un résultat tiré de Seumen-Tonou [81] qui permet de justifier une technique de changement de loi pour le temps de saut au moment du changement de zone. De plus, du fait des caractéristiques physiques très différentes des différentes zones et d'une plus grande taille du domaine spatial de calcul, la variance et les temps de simulations liés au calcul de u sont nettement plus grands que dans le modèle de base. Afin de pouvoir quand même utiliser notre approche, nous avons utilisé deux techniques permettant d'avoir une variance plus faible basées sur les propriétés physiques du milieu. La première a consisté à choisir le point de départ des trajectoires dans la zone où u est le plus grand, c'est à dire dans la zone la plus fissile (et même dans une partie de cette zone). Ceci a eu pour effet de mieux contrôler les trajectoires qui contribuent le plus au calcul de u pour t grand. Les résultats

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

n'étant pas complètement satisfaisants, nous avons utilisé alors une deuxième technique qui est d'initialiser la résolution du problème d'évolution avec une approximation (fournie par exemple par une autre méthode numérique) de la fonction propre associée à la valeur propre nulle. Ceci permet en effet comme précédemment de diminuer l'importance du terme lié aux autres valeurs propres dans le développement asymptotique nécessaire au calcul de la valeur propre principale. La combinaison de ces différentes techniques de réduction de variance a permis d'obtenir en bout de compte une précision de 4 ou 5 chiffres significatifs sur les valeurs propres principales et sur les facteurs critiques correspondants sur des problèmes considérés comme difficiles.

4.2.4 Une interprétation probabiliste dans le cas homogène

Nous avons utilisé jusqu'à présent la formule de Feynman-Kac comme un outil permettant d'obtenir une méthode numérique de calcul de la valeur propre principale de l'opérateur de transport neutronique dans un cadre très général. Nous allons maintenant prouver par des techniques probabilistes que l'on peut se servir de cette formule pour obtenir une représentation probabiliste ponctuelle de cette valeur propre dans le cadre de modèles homogènes vérifiant certaines hypothèses. On supposera toujours que le terme de gain $c(x,y)$ dans l'équation de transport est constant égal à ρ . Cette représentation va reposer sur le fait que la vitesse d'absorption du processus de transport par la frontière

$$\lim_{t \rightarrow \infty} \frac{1}{t} \log P(\tau_D^{x,y} > t)$$

est indépendante du point de départ (x,y) des trajectoires. On peut résumer ceci dans le théorème suivant dont la preuve est donnée dans [66] et qui s'inspire d'une démonstration de Kifer [47] pour les processus de diffusion.

Théorème 4.2.1. *On suppose qu'il existe T tel que si $t > T$, alors il existe une fonction positive et mesurable $p^D(t, (x,v), (x_1, v_1))$ vérifiant*

$$E_{x,v}[f(X_t, V_t) 1_{\tau_D > t}] = \int_V \int_D p^D(t, (x,v), (x_1, v_1)) f(x_1, v_1) dx_1 dv_1$$

pout toute fonction f mesurable et bornée. On suppose de plus qu'il existe un sous-ensemble A de $D \times V$ tel que

$$0 < \text{ess} \inf_{(x_1, v_1) \in A} p^D(t, (x,v), (x_1, v_1)) \leq \text{ess} \sup_{(x_1, v_1) \in A} p^D(t, (x,v), (x_1, v_1)) < \infty.$$

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

On suppose enfin qu'il existe une constante $\kappa > 0$ telle que

$$\int \int_A P_{x,v}(\tau_D > t) dx dv \geq \kappa \int \int_{A^c} P_{x,v}(\tau_D > t) dx dv$$

pour tout $t > 0$. On a alors pour tout $(x,v) \in D \times V$

$$\lim_{t \rightarrow +\infty} \frac{1}{t} \log(P_{x,v}(\tau_D > t)) = \beta_\rho.$$

De plus si il existe des constantes c_1 et c_2 telles que $0 < c_1 \leq u_0(x,v) \leq c_2 < \infty$ alors

$$\lim_{t \rightarrow +\infty} \frac{1}{t} \log(u(t,x,v)) = \beta_\rho + \rho.$$

Ce théorème prouve que le résultat classique pour les processus de Markov de l'existence d'une limite β_ρ quand $t \rightarrow +\infty$ pour

$$\frac{1}{t} \log \sup_{(x,v) \in D \times V} (P_{x,v}(\tau_D > t))$$

est en fait vrai aussi ponctuellement. Les hypothèses de ce théorème s'inspirent de celles de [47] mais en les adaptant aux processus de transport. Elles exigent l'existence d'une densité de transition à partir d'un certain temps T pour le processus de transport qui est à la fois bornée inférieurement et supérieurement sur un sous-ensemble de $D \times V$. Pour ce même processus, on doit de plus contrôler la valeur moyenne des probabilités de sortie de son complémentaire par la valeur moyenne des probabilités de sortie de lui-même.

Nous avons ensuite montré la validité de ces hypothèses sur un modèle 1D inspiré du modèle de Lehner-Wing [19,21] où l'espace des vitesses exclu un voisinage de zéro. Ce modèle s'écrit

$$\frac{\partial u}{\partial t} = -v \frac{\partial u}{\partial x} + c \left\{ \frac{1}{2(1-\alpha)} \int_{V_\alpha} u(x,v',t) dv' - u(x,v,t) \right\} + (c-1) u(x,v,t)$$

avec une condition initiale positive $(x,v,0) = u_0(x,v)$ et des condition aux limites absorbantes. Le domaine spatial est $]0,d[$ et le domaine des vitesses est $V_\alpha =]-1, -\alpha[\cup]\alpha, 1[$ avec $0 < \alpha < 1$ et $c > 0$. Le processus de transport (X_t, V_t) est solution de $\frac{dX_t}{dt} = -V_t$. La loi de la vitesse après une collision est la loi uniforme sur V_α et la loi conditionnelle du temps entre deux sauts sachant que $(X_t, V_t) = (x,v)$ avant la collision est une loi exponentielle de paramètre c . On a prouvé que les hypothèses du théorème précédent sont vérifiées pour l'ensemble $A = (\frac{d}{4}, \frac{3d}{4}) \times V_\alpha$.

En ce qui concerne la majoration de la densité, l'idée a été de se ramener à une situation familière dans le cas d'un domaine spatial infini où la loi initiale admet une densité bornée

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

par rapport à la mesure de Lebesgue. Dans ce cas précis, on peut en effet montrer que cette propriété est vraie à tout instant. Dans notre cas, la loi initiale est une mesure de Dirac au point (x,v) . Pour espérer obtenir une densité à un instant T , le processus (X_T, V_T) doit avoir subi un certain nombre de chocs avant cet instant. Grâce à la borne inférieure sur la norme des vitesses et à la bornitude du domaine, le choix du temps $T = \frac{3d}{\alpha}$ assure que le processus ait subi au moins trois chocs pour être encore en vie, ce qui est suffisant pour assurer l'existence et la majoration de la densité de (X_t, V_t) .

Pour la minoration, nous avons montré que l'on pouvait construire suffisamment de trajectoires qui en partant d'un point quelconque du domaine atteignent un point quelconque de A à un instant supérieur à T . Enfin, le contrôle de valeur moyenne de $P_{x,v}(\tau_D > t)$ sur A^c par la valeur moyenne de $P_{x,v}(\tau_D > t)$ sur A repose simplement sur le fait que des trajectoires qui partent d'une zone centrale ont une probabilité plus forte d'être encore en vie à la date t . Les démonstrations de ces différentes propriétés sont cependant assez techniques et sont détaillées dans [66].

4.3 Valeur propre principale du Laplacien

On considère maintenant le problème d'évolution

$$\frac{\partial u}{\partial t} = \frac{1}{2} \Delta u,$$

avec comme condition initiale $u(0,x) \equiv 1$ où $x \in D \subset \mathbb{R}^d$. La solution de cette équation est donnée par

$$u(x,t) = P(\tau_D^x > t)$$

où τ_D^x désigne le premier instant de sortie du mouvement Brownien B_t^x de l'ouvert D . La valeur propre principale λ_1 est alors directement reliée à la vitesse d'absorption de B_t^x par la frontière de l'ouvert D et on a $\forall x \in D$:

$$\lambda_1 = \lim_{t \rightarrow +\infty} \frac{1}{t} \log P(\tau_D^x > t).$$

Dans Freidlin [29], la même étude a été effectuée pour un opérateur elliptique très général avec des conditions aux limites de type Dirichlet dans un ouvert D borné. Les techniques numériques précédentes peuvent bien entendu s'adapter au calcul de λ_1 , mais cela exige une simulation non plus exacte mais approchée du mouvement Brownien qui remplace ici le processus de transport. Il existe un grand nombre de schémas de simulation de ce mouvement Brownien et du temps de sortie de celui-ci d'un domaine borné. La précision de ces schémas ainsi que leurs coûts sont cruciaux pour la validation de la méthode. Nous avons expérimenté

certaines de ces schémas dans le calcul de λ_1 que nous allons maintenant décrire rapidement.

4.3.1 Description des schémas

Le schéma d'Euler

Le schéma le plus classique dans la simulation du mouvement Brownien issu de x est le schéma d'Euler [4,48] de paramètre de discrétisation Δt qui s'écrit

$$X_0 = x, X_{n+1} = X_n + \sqrt{\Delta t} Y_n$$

où les Y_n sont des variables aléatoires Gaussiennes centrées et réduites. Pour calculer $P(\tau_D^x > t)$, on doit simuler numériquement τ_D^x . La version naïve consiste à arrêter la trajectoire dès que $X_{n+1} \in D^c$ et à approcher alors τ_D^x par exemple par $n \Delta t$ ou encore par une approximation légèrement meilleure basée sur les distances $d_n = d(X_n, \partial D)$ et $d_{n+1} = d(X_{n+1}, \partial D)$. Dans tous les cas, ce type d'approximation a seulement un ordre de convergence de $\sqrt{\Delta t}$ car il ne prend pas en compte l'erreur de simulation principale qui provient de la possibilité pour le mouvement Brownien de sortir du domaine puis d'y retourner entre deux pas de temps successifs. Il est possible de prendre mieux en compte cette possibilité en utilisant l'approximation dite de demi-espace [31,32] qui atteint un ordre de convergence de Δt . On ajoute simplement un tirage supplémentaire à chaque étape d'une variable aléatoire uniforme U_n qui oblige le mouvement à se stopper si

$$\exp\left(-\frac{2d_n d_{n+1}}{\Delta t}\right) > U_n.$$

Il est également possible de raffiner encore un peu cette approche en simulant une autre variable aléatoire au moment de la sortie du domaine [14] ce qui améliore un peu la précision mais ne change pas l'ordre de la méthode. Une autre méthode a été proposée par Gobet et Menozzi [36] où l'on diminue artificiellement la taille du domaine d'un rapport $(1 - C_0 \sqrt{\Delta t})$ avec $C_0 \simeq 0.58258$, en utilisant le schéma d'Euler naïf.

La marche sur les sphères

Les méthodes de simulation basées sur le schéma d'Euler sont également utilisables dans bien d'autres situations que la simulation d'un mouvement Brownien. En revanche, on dispose dans ce cas particulier d'autres schémas qui peuvent s'avérer plus efficaces que celui-ci. La marche sur les sphères [25,80] repose sur l'isotropie du mouvement Brownien et permet d'atteindre la frontière du domaine en faisant des grands sauts ce qui diminue considérablement le nombre de coups avant la sortie du domaine. Cette marche se fait d'une sphère à

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

une autre en tirant un point uniformément sur celle-ci et en choisissant le rayon de la sphère suivante le plus grand possible, égal à la distance de ce même point à la frontière du domaine. On arrête la marche quand on se trouve à une distance inférieure à ε de la frontière. L'erreur de simulation est un $O(\varepsilon)$ et le nombre moyen de coups avant la sortie est un $O(|\ln(\varepsilon)|)$. Il existe également des versions un peu plus sophistiquées de cette méthode qui permettent une absorption encore plus rapide par la frontière où le nombre moyen de coups avant la sortie est un $O(|\ln(|\ln(\varepsilon)|)|)$. Dans notre cas, on veut également simuler τ_D^x ce qui implique la simulation de la loi $Y(r)$ du temps de sortie d'une sphère de rayon r en partant de son centre. En utilisant un changement d'échelle, on peut prouver que $Y(r)$ et $r^2 Y(1)$ ont même loi de sorte qu'on a seulement à simuler $Y(1)$. La fonction de répartition $H(t)$ de $Y(1)$ en dimension 2 vaut par exemple [11]

$$H(t) = \sum_{k=1}^{\infty} \frac{2J_0(0)}{j_k J_1(j_k)} \exp\left(\frac{-j_k^2 t}{2}\right)$$

où J_0 et J_1 sont des fonctions de Bessel et les j_k sont les zéros positifs de J_0 . La méthode usuelle de simulation de $Y(1)$ conduit à l'inversion d'une série ce qui peut être difficile et coûteux. Pour accélérer cette simulation, on peut cependant tabuler $H(t)$ une fois pour toutes en utilisant soit son expression analytique, soit des simulations Monte-Carlo basées sur le schéma d'Euler avec un très grand nombre de trajectoires et un pas de discrétisation petit. Cette dernière technique est particulièrement efficace en dimension plus grande que 2.

La marche sur les rectangles

Dans bon nombre de situations pratiques, le domaine d'étude est, ou peut-être, approché par un polygone. On peut alors construire une simulation exacte du mouvement Brownien en utilisant non plus une marche sur des sphères mais une marche sur des carrés ou des rectangles (ou des parallélépipèdes en dimension plus grande que 2). On peut faire coller le rectangle à un des bords du domaine et on peut aussi choisir le point de départ du mouvement Brownien de n'importe quel point à l'intérieur de celui-ci. Ceci permet de diminuer plus encore qu'avec la marche sur les sphères le nombre de coups moyens avant la sortie du domaine et aussi de recouvrir le domaine par des rectangles avant même le début des simulations. En revanche, le temps de sortie et la position de sortie doivent être simulés simultanément et la loi de cette dernière n'est bien évidemment plus la loi uniforme sur le bord du domaine. On est capable d'effectuer cette simulation en dimension quelconque en se ramenant uniquement à des mouvements Browniens unidimensionnels. Une description détaillée de l'algorithme utilisé est faite dans [23]. La simulation des variables nécessaires à cet algorithme est encore plus complexe que pour la marche sur les sphères d'autant qu'on

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

ne peut faire qu'une tabulation partielle de leurs lois. Nous avons expérimenté des techniques de tirage d'importance à l'aide de variables aléatoires plus simples à simuler afin de réduire les temps de calculs et aussi la variance. Elles n'ont pas réellement abouti conduisant en fait à une augmentation de la variance qui détruisait une partie du bénéfice lié à la réduction des temps de calculs. Ces techniques sont cependant prometteuses et des tentatives d'améliorations sont en cours [24].

4.3.2 Résultats numériques

Dans [55], nous avons raffiné les critères d'ajustement du problème de moindres carrés utilisé dans [60,66] en sélectionnant les meilleurs estimateurs grâce au coefficient de corrélation et aux choix de fenêtres adéquates. Ces nouvelles techniques statistiques nous ont permis d'obtenir, en moins de 10 minutes de temps de calcul, une précision relative de l'ordre de 10^{-3} sur la valeur propre principale pour différents problèmes en dimensions deux à cinq. En ce qui concerne les schémas de simulation, le schéma d'Euler naïf doit être écarté pour le calcul de la valeur propre principale car il sous-estime beaucoup la vitesse d'absorption du mouvement Brownien par la frontière du domaine. L'utilisation de ce schéma a conduit à des erreurs de plus de 10 pour cent sur ce même calcul. Les autres schémas ont une précision analogue mais des comportements différents en terme de coûts de simulation. La marche sur les rectangles (en particulier dans le cas de domaines polygonaux) ne demande qu'un très petit nombre de coups pour sortir du domaine, la marche sur les sphères de l'ordre de 5 à 10 fois plus, le schéma d'Euler de 100 à 1000 fois plus. En revanche, la complexité en termes de simulation de variables aléatoires se fait dans l'ordre inverse. Un bon compromis semble donc d'utiliser la marche sur les sphères même si un travail numérique encore plus poussé pourrait faire pencher la balance en faveur de la marche sur les rectangles [24].

4.4 Réduction de variance à l'aide de branchements

4.4.1 Description de la méthode

Dans les cas des opérateurs de transports homogènes et dans le cas de l'opérateur Laplacien, nous avons réussi à nous affranchir d'une partie des problèmes liés à la gestion des événements rares dans le calcul de $P(\tau_D^x > t)$ grâce à l'utilisation du développement spectral. Ce développement permet en effet de ne pas avoir à considérer des temps t trop grands et donc à calculer la probabilités d'événements très rares. En posant $p_1 = P(\tau_D^x > t)$, $p_2 = P(\tau_D^x > t)$ et $s_1 = \sqrt{p_1(1-p_1)}$, $s_2 = \sqrt{p_2(1-p_2)}$, on peut cependant voir sur l'intervalle de confiance

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

à au moins 90 pour cent [55]

$$\frac{\log(\frac{p_2}{p_1})}{t_2 - t_1} - \frac{1.96N^{-\frac{1}{2}}}{t_2 - t_1}(\frac{s_1}{\bar{p}_1} + \frac{s_2}{\bar{p}_2}) \leq \alpha(t_1, t_2) \leq \frac{\log(\frac{p_2}{p_1})}{t_2 - t_1} + \frac{1.96N^{-\frac{1}{2}}}{t_2 - t_1}(\frac{s_1}{\bar{p}_1} + \frac{s_2}{\bar{p}_2})$$

pour l'estimateur $\alpha(t_1, t_2)$ de λ_1 par la méthode dite d'interpolation que la gestion des événements rares n'est pas parfaite. En effet, des facteurs en $\frac{1}{\sqrt{p_1}}$ et $\frac{1}{\sqrt{p_2}}$ apparaissent dans les bornes de cet intervalle où $\bar{p}_1$ et $\bar{p}_2$ sont les estimateurs de p_1 et p_2 . Des techniques récentes [17,26,50] basées sur des processus de branchement permettent de s'affranchir de ce problème en remplaçant le calcul Monte Carlo de la probabilité d'un événement rare par le calcul de produits d'événements qui le sont moins. Dans notre cas [56], on procède de la manière suivante pour le calcul de $P_x(\tau_D > T)$. On fait partir N trajectoires issus de x et on stocke à un premier instant intermédiaire T_1 la position et la vitesse (dans le cas des processus de transport) des particules qui sont encore en vie à cet instant. A la date T_1 , on fait ensuite repartir N particules tirés suivant la loi empirique π_{T_1} des particules encore en vie à cette même date. On peut utiliser plusieurs tranches intermédiaires $T_1, T_2, \dots, T_k$ suivant la rareté de l'événement que l'on considère. Une discussion complète sur l'optimisation du choix des tranches en termes de variance en fonction des différents paramètres est faite dans [50]. Un bon choix consiste à choisir des tranches pour lesquelles la probabilité de passer de l'une à l'autre est de 0.2. Le nombre de tranches est alors directement relié à la probabilité de l'événement que l'on considère. Bien entendu, cela peut poser certaines difficultés pratiques car on ne connaît pas les instants T_k qui réalisent ce choix ni la probabilité de l'événement considéré. Grâce à l'utilisation du développement spectral, nous n'avons besoin que de deux ou trois tranches et les instants T_k sont estimés à l'aide de simulations Monte Carlo utilisant un petit nombre de trajectoires. Le calcul de $P_x(\tau_D > T)$ s'obtient en faisant le produit des probabilités de passage d'une tranche à l'autre. En effet, on a par exemple grâce à la propriété de Markov

$$P_x(\tau_D > T) = P_x(\tau_D > T_1) P_{\pi_{T_1}}(\tau_D > T - T_1)$$

dans le cas d'une seule tranche intermédiaire.

4.4.2 Application au calcul de fonctions propres

On peut également se demander ce que représente les particules encore en vie après plusieurs branchements successifs. Pour cela, on s'intéresse à la loi conditionnelle de X_t

Chapitre 4. Approximation de la valeur propre principale de différents opérateurs linéaires

sachant que $\tau_D > t$ en écrivant que

$$E_x[u_0(X_t)/\tau_D > t] = \frac{E_x[u_0(X_t); \tau_D > t]}{E_x[1; \tau_D > t]} \simeq \frac{\langle u_0, \varphi_1^* \rangle \varphi_1(x) \exp(\lambda_1 t)}{\langle 1, \varphi_1^* \rangle \varphi_1(x) \exp(\lambda_1 t)} = \frac{\langle u_0, \varphi_1^* \rangle}{\langle 1, \varphi_1^* \rangle}$$

pour un temps t suffisamment grand. Cette formule montre donc que cette loi conditionnelle tend vers $\frac{\varphi_1^*}{\langle 1, \varphi_1^* \rangle}$ quand t est grand. Cela permet donc d'obtenir une approximation de la fonction propre de l'adjoint de l'opérateur que l'on considère à l'aide de particules. On peut soit construire un histogramme pour obtenir cette approximation soit utiliser une convolution avec par exemple un noyau Gaussien. Dans le cas d'un opérateur auto-adjoint comme le Laplacien, on a donc directement une approximation de φ_1 . Cette approximation peut aussi s'obtenir dans beaucoup d'autres cas en utilisant la méthode sur l'opérateur adjoint. Tout ceci est décrit en détail dans [56] ainsi que de nombreux tests numériques sur des opérateurs de transport et de diffusion. Ces tests ont montré que l'utilisation de ces nouvelles techniques de branchements permettent de diviser par 100 le nombre de trajectoires nécessaires pour obtenir la valeur propre principale avec une précision donnée. De plus, une bonne précision a aussi été obtenue sur la fonction propre liée à cette valeur propre principale.

4.5 Conclusion

Les techniques présentées ici ont permis de calculer numériquement de manière précise la valeur propre principale d'opérateurs de transport neutronique et de l'opérateur de Laplace. Elles ont également permis de donner une représentation probabiliste de la valeur propre principale de l'opérateur de transport neutronique pour un modèle homogène particulier. La méthode basée sur les processus de branchements a permis d'améliorer la précision sur le calcul de la valeur propre principale mais a surtout fourni une méthode de calcul de la fonction propre liée à l'adjoint de l'opérateur considéré. De nombreuses pistes restent encore à explorer aussi bien d'un point de vue théorique que numérique comme par exemple l'extension de la représentation probabiliste à des modèles plus compliqués ou encore le calcul numérique de la seconde valeur propre dominante de l'opérateur.

Bibliographie

- [1] J. A. ACEBRON, M. P. BUSICO, P. LANUCARA, R. SPIGLER, Probabilistically induced domain decomposition methods for elliptic boundary-value problems. *Journal of Computational Physics*. 210, no.2, pp. 421-438, 2005.
- [2] J. A. ACEBRON, M. P. BUSICO, P. LANUCARA, R. SPIGLER, Domain decomposition solution of elliptic boundary-value problems via Monte Carlo and quasi-Monte Carlo methods. *SIAM journal of Scientific Computing*, 27, no.2, pp. 440-457, 2005.
- [3] E. ATANASSOV, I. T. DIMOV, A new optimal Monte Carlo method for calculating integral of smooth functions, *Monte Carlo Methods and Appl.*, Vol. 5, No. 2, pp. 149-167, 1999.
- [4] V. BALLY, D. TALAY, The law of the Euler scheme for stochastic differential equations (I) : Convergence rate of the distribution function. *Probability theory and related fields*, 104 : 43-60, 1995.
- [5] C. BARDOS, R. SANTOS and R. SENTIS. Diffusion approximation and computation of the critical size. *Transactions of the american mathematical society*, Volume 84, Number 2, August 1984.
- [6] M. BECKERS, A. HAEGMANS, Transformations of integrands for lattice rules, in *Numerical Integration-Recent Developpements, Software and applications*, T. O. Espelid, A.Genx, Eds., Kluwer, Dordrecht, the Netherlands, pp. 329-340, 1992.
- [7] A. BENVENISTE, M. METIVIER, P. PRIOURET, *Algorithmes stochastiques et approximations stochastiques*, Masson, Paris, 1987.
- [8] C. BERNARDI, Y. MADAY, *Approximations spectrales de problèmes aux limites elliptiques*. Springer-Verlag, 1992.
- [9] A. BJORK, *Numerical methods for least squares problems*, SIAM, 1996.
- [10] T.E. BOOTH, Exact Monte Carlo solution of elliptic partial differential equation, *J. Comput. Phys.* 47 (1982) 396-404.
- [11] A. BORODIN, P. SALMINEN, *Handbook of Brownian motion : facts and formulae*. Boston, Ma : Birkhauser, 1996.
- [12] N. BOULEAU. *Probabilités de l'ingénieur*. Hermann, 1986.

Bibliographie

- [13] D. BROCKWAY, P. SORAN, P. WHALEN, Monte Carlo eigenvalue calculation, in Monte Carlo methods and applications in neutronics photonics and statistical physics. R. ALCOUFFE et al (Eds), 378-384, Springer-Verlag, 1985.
- [14] F. BUCHMANN, Simulation of stopped diffusions, J. Comput. Phys. 202, no.2, pp 446-462, 2005.
- [15] J. BUSSAC et P.REUSS. Traité de neutronique. Hermann, Paris, 1978.
- [16] C. CANUTO, M. Y. HUSSAINI, A. QUATERONI, T. A. ZANG, Spectral methods in fluid dynamics. Springer-Verlag, 1988.
- [17] F. CEROU, P. DEL MORAL, F. LEGLAND, P. LEZAUD, Genetic genealogical models in rare event analysis, ALEA Lat. Am. J. Probab. Math. Stat. 1, pp. 181-203, 2006.
- [18] I. COULIBALY, C. LÉCOT, A quasi-randomized Runge-Kutta method. Mathematics of Computation, 68, no.226, pp. 651-659, 1999.
- [19] E. B. DAHL and N. G. SJOSTRAND. Eigenvalue spectrum of multiplying slabs and spheres for monoenergetic neutrons with anisotropic scattering, 1978.
- [20] R. DAUTRAY et al. Méthodes probabilistes pour les équations de la physique. Eyrolles, 1989.
- [21] R. DAUTRAY, J.L. LIONS. Analyse mathématique et calcul numérique pour les sciences et les techniques. Masson, 1988.
- [22] P. DAVIS, P. RABINOWITZ, Methods of numerical integration, second edition, Comp. Sci. and applied Math., Academic Press, 1984.
- [23] M. DEACONU, A. LEJAY, A random walk on rectangles algorithm. Methodol. Comput. Appl. Proba., 8, no.1, pp 135-151, 2006.
- [24] M. DEACONU, A. LEJAY, Simulating SDE using the importance sampling paradigm. Soumis.
- [25] J.M DE LAURENTIS, L.A ROMERO, A Monte Carlo method for Poisson's equation, J. Comput. Phys. 90, 1990.
- [26] P. DEL MORAL, J. GARNIER, Genealogical particle analysis of rare events, Ann. Appl. Probab, 15, no 4, 2496-2534, 2005.
- [27] C. DELUIGI, S. DUMONT, S. MAIRE, Sparse polynomial approximations of the Poisson equation over hypercubes, Preprint du laboratoire ANAM 2007(2).
- [28] C. FARHAT, F.-X. ROUX, Implicit parallel processing in structural mechanics, IACM, Computational Mechanics Advances, Vol. 2 No.1, North-Holland, 1994.
- [29] M. FREIDLIN. Functional Integration and Partial Differential Equations. Princeton University Press, 1985.
- [30] A. FRIEDMAN. Stochastic Differential Equations and Applications. Volume 2, Academic Press, 1976.

Bibliographie

- [31] E. GOBET, Weak approximations of killed diffusions using Euler schemes. Stochastic processes and their applications, Vol 87, pp 167-197, 2000.
- [32] E. GOBET, Euler schemes and half-space approximations for the simulation of diffusion in a domain, ESAIM Probability and Statistics, Vol 5, pp 261-297, 2001.
- [33] E. GOBET, S. MAIRE, A spectral Monte Carlo method for the Poisson equation. Monte Carlo Methods and applications, Vol. 10, No. 3-4, pp. 275-285, 2004.
- [34] E. GOBET, S. MAIRE, Sequential control variates for functionals of Markov processes, SIAM Journal on Numerical Analysis, 43, no.3, pp. 1256-1275, 2005.
- [35] E. GOBET, S. MAIRE, Sequential Monte Carlo domain decomposition for the Poisson equation, Proceedings of the 17 imacs World congress, Scientific Computation, Applied Mathematics and Simulation 2005.
- [36] E. GOBET, S. MENOZZI. Exact approximation rate of killed hypoelliptic diffusions using the discrete Euler scheme, Stochastic Process. Appl, 112, no. 2, pp 201-223, 2004.
- [37] J. HALTON, Sequential Monte Carlo, Proc. Camb. Phil. Soc, Vol. 58, pp 57-78, 1962.
- [38] P. HELLUY, S. MAIRE, P. RAVEL, Intégration numérique d'ordre élevé de fonctions régulières ou singulières sur un intervalle. C. R. Acad. Sci. Paris Sér. I Math. 327, no.9, pp. 843-848, 1998.
- [39] P. HELLUY, S. MAIRE, P. RAVEL, Higher order numeric quadratures for regular or singular functions, applications for the Helmholtz integral equations, ASME design engineering technical conferences, 1999.
- [40] F. J. HICKERNEL, My dream quadrature rule, Journal of complexity, Vol.19, pp. 420-427, 2003.
- [41] C. HWANG, M. MASCAGNI, J.A. GIVENS. A Feynman-Kac path-integral implementation for Poisson's equation using an h-conditioned Green function. Mathematics and computers in simulation 62, 2003.
- [42] M. IRI, S. MORIGUTI, Y. TAKAZAWA, On a certain quadrature formula (in japanese), Kokyuroku of Res., Inst. for Math. Sci., Kyoto Univ., Vol. 91, pp. 82-118, 1970.
- [43] S. JOE, I. SLOAN, Imbedded lattice rules for multidimensional integration, SIAM J. Numer. Anal. Vol.29, No. 4 pp. 1119-1135, 1992.
- [44] M. KAC. On some connections between probability theory and differential and integral equations, Proceedings of the second Berkeley symposium, 1950, pp. 189-215.
- [45] M. H. KALOS. Equation du transport. Cours à l'école CEA-EDF-INRIA sur les méthodes de Monte-Carlo, INRIA Reports, 1981.
- [46] M. H. KALOS, P. A. WHITLOCK, Monte Carlo Methods, John Wiley & Sons, 1986.
- [47] Y. KIFER. Random perturbations of dynamical systems. Birkhauser Boston, 1988.

Bibliographie

- [48] P. KLOEDEN, E. PLATEN, Numerical solution of stochastic differential equations. Applications of Mathematics, Springer-Verlag, Berlin, 1992.
- [49] A. R. KROMMER, C. W. UEBERHUBER. Computational integration. SIAM, 1998.
- [50] A. LAGNOUX, Rare events simulation, Probab. Engrg. Inform. Sci. 20, no 1, pp. 45-66, 2006.
- [51] B. LAPEYRE. E. PARDOUX, R. SENTIS, Méthodes de Monte-Carlo pour les équations de transport et de diffusion. Springer-Verlag, 1998.
- [52] C. LÉCOT, I. COULIBALY, A quasi-Monte Carlo scheme using nets for a linear Boltzmann equation. SIAM journal on numerical analysis, 35, no. 1, pp. 51-70, 1998.
- [53] C. LÉCOT, F. EL KHETTABI, Quasi-Monte Carlo simulation of diffusion. Dagstuhl Seminar on Algorithms and Complexity for continuous problems (1998). Journal of complexity 15, no.3, pp. 342-359, 1999.
- [54] C. LÉCOT, B. TUFFIN, Comparison of quasi-Monte Carlo methods for simulation of Markov chains. Monte Carlo methods and applications, Vol. 10(3-4), pp. 377-384, 2004.
- [55] A. LEJAY, S. MAIRE, Computing the principal eigenvalue of the Laplace operator by a stochastic method. Mathematics and computers in simulation 73, pp. 351-363, 2007.
- [56] A. LEJAY, S. MAIRE, Computing the first eigenelements of some linear operators using a branching Monte Carlo method. Preprint 2007(6) du laboratoire ANAM, soumis.
- [57] G. P. LEPAGE, A New Algorithm for Adaptative Multidimensional Integration, Journal of Computational Physics, Vol 27, pp. 192-203, 1978.
- [58] P. LE TALLEC, Domain decomposition methods in computational mechanics, IACM, Computational Mechanics Advances, Vol. 1 No.2, North-Holland, 1994.
- [59] E. LEWIS. Numerical methods for deterministic transport computations. Cours à l'école CEA-EDF-INRIA sur les méthodes de Monte-Carlo, INRIA Reports, 1981.
- [60] S. MAIRE, Réduction de variance pour l'intégration numérique et pour le calcul critique en transport neutronique, Thèse de doctorat, 2001.
- [61] S. MAIRE, Reducing variance using iterated control variates, The Journal of Statistical Computation and Simulation, Vol. 73(1), pp. 1-29. 2003.
- [62] S. MAIRE , Un algorithme probabiliste de calcul d'approximations polynômiales sur un hypercube, C. R. Acad. Sci. Paris, Ser. I 336, pp. 185-190, 2003
- [63] S. MAIRE, An iterative computation of approximations on Korobov-like spaces. Journal of Computational and Applied Mathematics, 157, pp. 261-281, 2003.
- [64] S. MAIRE, Polynomial Approximations of multivariate smooth functions from quasi-random data, Statistics and Computing, 14, pp. 333-336, 2004.
- [65] S. MAIRE, C. DE LUIGI, Quasi-Monte Carlo quadratures for multivariate smooth functions, Applied Numerical mathematics, 56, pp. 146-162, 2006.

Bibliographie

- [66] S. MAIRE, D. TALAY, On a Monte Carlo method for neutron transport criticality computations, *IMA Journal of Numerical Analysis* 26, no 4, pp. 657-685, 2006.
- [67] S. MAIRE, E. TANRE, Some new simulations schemes for the evaluation of Feynman-Kac representations. En préparation.
- [68] J. MIKA. Existence and uniqueness of the solution to the critical problem in neutron transport theory, *Studia Mathematica*, 37, pp 213-225, 1971.
- [69] M. MOKHTAR-KHARROUBI. *Mathematical Topics in Neutron Transport Theory, New aspects*, World Scientific, 1997.
- [70] N. NEWTON, Variance reduction for simulated diffusions, *SIAM journal on applied mathematics* 54, no 6, pp. 1780-1805, 1994.
- [71] H. NIEDERREITER, Quasi-Monte Carlo methods and pseudorandom numbers, *Bull. Amer. Math. Soc.* 84, pp. 957-1041, 1978.
- [72] E. NOVAK, K. RITTER, High dimensional integration of smooth functions over cubes, *Numerische Mathematik*, 75, pp. 79-97, 1996.
- [73] G. PAGES, A space vector quantization for numerical Integration, *Journal of computational and applied mathematics*, 89, pp. 1-38, 1997.
- [74] G. PAGES, J. PRINTEMPS, Optimal quadratic quantization for numerics: the Gaussian case, *Monte Carlo methods and applications*, Vol 9 (2), pp 135-166, 2003.
- [75] E. PEIRANO, D. TALAY, Domain decomposition by stochastic methods, *Domain decomposition in science and engineering*, I. Herrera, D. Keyes, O. Widlund, R. Yates (Eds), 2003.
- [76] T. VON PETERSDORFF, C. SCHWAB, Numerical solution of parabolic equations in high dimensions. *M2AN Math. Model. Numer. Anal.* 38, no.1, pp. 93-127, 2004.
- [77] A. PLANCHARD. *Méthodes mathématiques en neutronique*. Eyrolles, 1995.
- [78] W. H. PRESS, G. R. FARRAR, Recursive Stratified Sampling for Multidimensional Monte Carlo Integration, *Computer in Physics*, vol. 4, pp. 190-195, 1990.
- [79] W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING, B. P. FLANNERY, *Numerical recipes in FORTRAN*. Cambridge University Press. 1996.
- [80] K. SABELFELD, *Monte Carlo methods in boundary value problems*. Springer, Berlin, 1991.
- [81] P. SEUMEN-TONOU. *Méthodes numériques probabilistes pour la résolution d'équations du transport et pour l'évaluation d'options exotiques*. Thèse de doctorat 1997.
- [82] I. H. SLOAN, P. J. KACHOYAN, Lattice methods for multiple integration: Theory, error analysis and examples, *SIAM J. Numer. Anal.* 24, pp. 116-128, 1987.
- [83] S. A. SMOLIAK, Quadrature and interpolation formulas for tensor products of certain classes of functions, *Soviet Math. Dokl.* 4, pp. 240-243, 1963.

Bibliographie

- [84] I. M. SOBOL, The distributions of points in a cube and the approximate evaluation of integrals, Zh. Vychisl. Mat. i. Mat. Fiz, pp. 784-802, 1967.
- [85] I. VIDAV. Existence and Uniqueness of Nonnegative Eigenfunctions of the Boltzmann Operator. Journal of Mathematical Analysis and Applications 22, pp. 144-155, 1968.
- [86] X. WARIN. Méthodes déterministes de résolution de l'équation intégral-différentiel du transport neutronique. Note technique HI-72/93/066, EDF, 1993.